

# Revisiting natural actor-critics with value function approximation

Journées Francophones de Planification, Décision et  
Apprentissage pour la conduite de systèmes

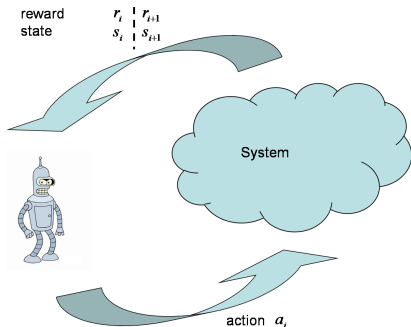
Matthieu GEIST and Olivier PIETQUIN

matthieu.geist@supelec.fr

2 juin 2010



- 1 Background
  - Paradigm
  - MDP and Bellman equations
  - Actor, Critic and actor-critic
- 2 Natural actor-critics
  - Policy gradient
  - Policy gradient with FA
  - Natural policy gradient with FA
  - Deriving a critic
- 3 Revisiting Natural actor-critics
  - Semi-compatible approximation
  - Revisiting (natural) policy gradient with FA
  - Deriving new algorithms
- 4 Illustration and conclusion
  - Preliminary results
  - Conclusion and perspectives



## Objective

Optimal control of a dynamic system :

- optimality : max. discounted cumulative reward
- from interactions  
( $s_i, a_i, r_i, s_{i+1}$ )
- without knowing the model

## Markov Decision Process

$$\text{MDP} = \{S, A, P, R, \gamma\}$$

$S$  state space,  $A$  action space,  $\gamma$  discount factor

$$P : s, a \in S \times A \rightarrow p(\cdot | s, a) \in \mathcal{P}(S)$$

$$R : s, a, s' \in S \times A \times S \rightarrow r = R(s, a, s') \in \mathbb{R}$$

## Markov Decision Process

$$\text{MDP} = \{S, A, P, R, \gamma\}$$

$S$  state space,  $A$  action space,  $\gamma$  discount factor

$$P : s, a \in S \times A \rightarrow p(\cdot | s, a) \in \mathcal{P}(S)$$

$$R : s, a, s' \in S \times A \times S \rightarrow r = R(s, a, s') \in \mathbb{R}$$

## Policy

$$\pi : s \in S \rightarrow \pi(\cdot | s) \in \mathcal{P}(A)$$

## Markov Decision Process

$$\text{MDP} = \{S, A, P, R, \gamma\}$$

$S$  state space,  $A$  action space,  $\gamma$  discount factor

$$P : s, a \in S \times A \rightarrow p(\cdot | s, a) \in \mathcal{P}(S)$$

$$R : s, a, s' \in S \times A \times S \rightarrow r = R(s, a, s') \in \mathbb{R}$$

## Policy

$$\pi : s \in S \rightarrow \pi(\cdot | s) \in \mathcal{P}(A)$$

## (State-action) value function

$$Q^\pi(s, a) = E[\sum_{i=0}^{\infty} \gamma^i r_i | s_0 = s, a_0 = a, \pi]$$

## Markov Decision Process

$$\text{MDP} = \{S, A, P, R, \gamma\}$$

$S$  state space,  $A$  action space,  $\gamma$  discount factor

$$P : s, a \in S \times A \rightarrow p(\cdot | s, a) \in \mathcal{P}(S)$$

$$R : s, a, s' \in S \times A \times S \rightarrow r = R(s, a, s') \in \mathbb{R}$$

## Policy

$$\pi : s \in S \rightarrow \pi(\cdot | s) \in \mathcal{P}(A)$$

## (State-action) value function

$$Q^\pi(s, a) = E[\sum_{i=0}^{\infty} \gamma^i r_i | s_0 = s, a_0 = a, \pi]$$

## Optimal policy

$$\pi^* = \operatorname{argmax}_{\pi} Q^\pi$$

$$\pi^*(s) = \operatorname{argmax}_{a \in A} Q^*(s, a)$$

## Bellman evaluation equation

$$Q^\pi(s, a) = E_{s', a' | s, a, \pi} [R(s, a, s') + \gamma Q^\pi(s', a')]$$

$$Q^\pi = T^\pi Q^\pi$$

⇒ policy iteration framework

## Bellman evaluation equation

$$Q^\pi(s, a) = E_{s', a' | s, a, \pi} [R(s, a, s') + \gamma Q^\pi(s', a')]$$

$$Q^\pi = T^\pi Q^\pi$$

⇒ policy iteration framework

## Bellman optimality equation

$$Q^*(s, a) = E_{s' | s, a} [R(s, a, s') + \gamma \max_{b \in A} Q^*(s', b)]$$

$$Q^* = T^* Q^*$$

⇒ value iteration framework

## Bellman evaluation equation

$$Q^\pi(s, a) = E_{s', a' | s, a, \pi} [R(s, a, s') + \gamma Q^\pi(s', a')]$$

$$Q^\pi = T^\pi Q^\pi$$

⇒ policy iteration framework

## Bellman optimality equation

$$Q^*(s, a) = E_{s' | s, a} [R(s, a, s') + \gamma \max_{b \in A} Q^*(s', b)]$$

$$Q^* = T^* Q^*$$

⇒ value iteration framework

## Goal of RL

- find the optimal policy  $\pi^*$ , or at least a near optimal policy
- here we focus on finding an optimal policy from a given initial state  $s_0$  :

$$\rho(\pi) = E[\sum_{i=0}^{\infty} \gamma^i r_i | s_0, \pi]$$

## Critic architecture

- $\pi^*$  is greedy resp. to  $Q^*$
- estimate  $\hat{Q}^*$ , define  $\hat{\pi}^*$  as
$$\hat{\pi}^*(s) = \operatorname{argmax}_{a \in A} \hat{Q}^*(s, a)$$
- **no structure for the policy**

## Critic architecture

- $\pi^*$  is greedy resp. to  $Q^*$
- estimate  $\hat{Q}^*$ , define  $\hat{\pi}^*$  as
$$\hat{\pi}^*(s) = \operatorname{argmax}_{a \in A} \hat{Q}^*(s, a)$$
- **no structure for the policy**

## Actor architecture

- try directly to maximize  $\rho(\pi)$ , without maintaining a structure for  $Q^\pi$
- *e.g.*, gradient ascent
$$\pi \leftarrow \pi + \alpha \nabla_{\pi} \rho(\pi)$$
- **no structure for the value function**

## Critic architecture

- $\pi^*$  is greedy resp. to  $Q^*$
- estimate  $\hat{Q}^*$ , define  $\hat{\pi}^*$  as  

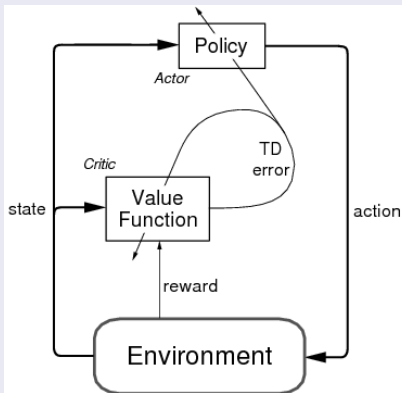
$$\hat{\pi}^*(s) = \operatorname{argmax}_{a \in A} \hat{Q}^*(s, a)$$
- **no structure for the policy**

## Actor architecture

- try directly to maximize  $\rho(\pi)$ , without maintaining a structure for  $Q^\pi$
- e.g., gradient ascent  

$$\pi \leftarrow \pi + \alpha \nabla_{\pi} \rho(\pi)$$
- **no structure for the value function**

## Actor-critic architecture



- **a structure for both the policy and the value function**
- **structures are interacting**

- 1 Background
  - Paradigm
  - MDP and Bellman equations
  - Actor, Critic and actor-critic
- 2 **Natural actor-critics**
  - **Policy gradient**
  - **Policy gradient with FA**
  - **Natural policy gradient with FA**
  - **Deriving a critic**
- 3 Revisiting Natural actor-critics
  - Semi-compatible approximation
  - Revisiting (natural) policy gradient with FA
  - Deriving new algorithms
- 4 Illustration and conclusion
  - Preliminary results
  - Conclusion and perspectives

- assume that the policy  $\pi$  is parametrized by  $\omega \in \mathbb{R}^p$

$$\text{e.g., } \pi_{\omega}(a|s) = \frac{\exp(\omega^T \phi(s,a))}{\sum_{b \in A} \exp(\omega^T \phi(s,b))}$$

- assume that the policy  $\pi$  is parametrized by  $\omega \in \mathbb{R}^p$   
e.g.,  $\pi_\omega(a|s) = \frac{\exp(\omega^T \phi(s,a))}{\sum_{b \in A} \exp(\omega^T \phi(s,b))}$
- a natural idea is to correct  $\pi_\omega$  according to a gradient ascent  
$$\omega_i = \omega_{i-1} + \alpha_i \nabla_\omega \rho(\pi_{\omega_{i-1}})$$

- assume that the policy  $\pi$  is parametrized by  $\omega \in \mathbb{R}^p$   
e.g.,  $\pi_\omega(a|s) = \frac{\exp(\omega^T \phi(s,a))}{\sum_{b \in A} \exp(\omega^T \phi(s,b))}$
- a natural idea is to correct  $\pi_\omega$  according to a gradient ascent  
$$\omega_i = \omega_{i-1} + \alpha_i \nabla_\omega \rho(\pi_{\omega_{i-1}})$$
- how to express  $\nabla_\omega \rho(\pi_\omega) = \nabla_\omega (E[\sum_{i=0}^{\infty} \gamma^i r_i | \mathbf{s}_0, \pi_\omega])$  ?

- assume that the policy  $\pi$  is parametrized by  $\omega \in \mathbb{R}^p$   
*e.g.*,  $\pi_\omega(a|s) = \frac{\exp(\omega^T \phi(s,a))}{\sum_{b \in A} \exp(\omega^T \phi(s,b))}$
- a natural idea is to correct  $\pi_\omega$  according to a gradient ascent  
 $\omega_i = \omega_{i-1} + \alpha_i \nabla_\omega \rho(\pi_{\omega_{i-1}})$
- how to express  $\nabla_\omega \rho(\pi_\omega) = \nabla_\omega (E[\sum_{i=0}^{\infty} \gamma^i r_i | s_0, \pi_\omega])$  ?

## Policy Gradient Theorem

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in A} Q^{\pi_\omega}(s, a) \nabla_\omega \pi_\omega(a|s)$$

with  $d^\pi$  the discounted weighting of states encountered

$$d^\pi(s) = \sum_{i=0}^{\infty} \gamma^i p(s_i = s | s_0, \pi)$$

- PG theorem :

$$\nabla_{\omega} \rho(\pi_{\omega}) = \sum_{s \in \mathcal{S}} d^{\pi}(s) \sum_{a \in \mathcal{A}} Q^{\pi_{\omega}}(s, a) \nabla_{\omega} \pi_{\omega}(a|s)$$

- PG theorem : replace  $Q^{\pi_\omega}$  by  $\hat{Q}_\theta$  ?

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in \mathcal{A}} \hat{Q}_\theta(s, a) \nabla_\omega \pi_\omega(a|s)$$

- PG theorem : replace  $Q^{\pi_\omega}$  by  $\hat{Q}_\theta$  ?

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in A} \hat{Q}_\theta(s, a) \nabla_\omega \pi_\omega(a|s)$$

- the response is YES if

- PG theorem : replace  $Q^{\pi_\omega}$  by  $\hat{Q}_\theta$  ?

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in \mathcal{A}} \hat{Q}_\theta(s, a) \nabla_\omega \pi_\omega(a|s)$$

- the response is YES if

- $\hat{Q}_\theta$  is a good approximation :

$$E_{s,a|d^{\pi_\omega}, \pi_\omega} [(Q^{\pi_\omega}(s, a) - \hat{Q}_\theta(s, a)) \nabla_\theta \hat{Q}_\theta(s, a)] = 0$$

- PG theorem : replace  $Q^{\pi_\omega}$  by  $\hat{Q}_\theta$  ?

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in \mathcal{A}} \hat{Q}_\theta(s, a) \nabla_\omega \pi_\omega(a|s)$$

- the response is **YES** if

- $\hat{Q}_\theta$  is a **good approximation** :

$$E_{s,a|d^{\pi_\omega}, \pi_\omega} [(Q^{\pi_\omega}(s, a) - \hat{Q}_\theta(s, a)) \nabla_\theta \hat{Q}_\theta(s, a)] = 0$$

- the **parametrization is compatible** :

$$\nabla_\theta \hat{Q}_\theta(s, a) = \nabla_\omega \ln \pi_\omega(a|s)$$

- PG theorem : replace  $Q^{\pi_\omega}$  by  $\hat{Q}_\theta$  ?

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in \mathcal{A}} \hat{Q}_\theta(s, a) \nabla_\omega \pi_\omega(a|s)$$

- the response is **YES** if

- $\hat{Q}_\theta$  is a **good approximation** :

$$E_{s,a|d^{\pi_\omega}, \pi_\omega} [(Q^{\pi_\omega}(s, a) - \hat{Q}_\theta(s, a)) \nabla_\theta \hat{Q}_\theta(s, a)] = 0$$

- the **parametrization is compatible** :

$$\nabla_\theta \hat{Q}_\theta(s, a) = \nabla_\omega \ln \pi_\omega(a|s)$$

## Policy gradient with function approximation

Under these assumptions :

$$\nabla_\omega \rho(\pi_\omega) = \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in \mathcal{A}} \hat{Q}_\theta(s, a) \nabla_\omega \pi_\omega(a|s)$$

- natural policy gradient : replace the gradient by a natural gradient

- natural policy gradient : replace the gradient by a natural gradient
- natural gradient : gradient pre-multiplied by the inverse of the Fisher information matrix :

$$\tilde{\nabla} \rho(\pi_{\omega}) = \mathbf{G}^{-1}(\omega) \nabla \rho(\pi_{\omega})$$

- natural policy gradient : replace the gradient by a natural gradient
- natural gradient : gradient pre-multiplied by the inverse of the Fisher information matrix :

$$\tilde{\nabla} \rho(\pi_\omega) = \mathbf{G}^{-1}(\omega) \nabla \rho(\pi_\omega)$$

- the Fisher information matrix is equal to :

$$\mathbf{G}(\omega) = E_{s,a|d^{\pi_\omega}, \pi_\omega} [\nabla_\omega \ln \pi_\omega(a|s) \nabla_\omega^T \ln \pi_\omega(a|s)]$$

- natural policy gradient : replace the gradient by a natural gradient
- natural gradient : gradient pre-multiplied by the inverse of the Fisher information matrix :  

$$\tilde{\nabla} \rho(\pi_\omega) = \mathbf{G}^{-1}(\omega) \nabla \rho(\pi_\omega)$$
- the Fisher information matrix is equal to :  

$$\mathbf{G}(\omega) = E_{s, a | d^{\pi_\omega, \pi_\omega}} [\nabla_\omega \ln \pi_\omega(a|s) \nabla_\omega^T \ln \pi_\omega(a|s)]$$

### Natural policy gradient with FA

under the same assumptions as before :

$$\tilde{\nabla} \rho(\pi_\omega) = \theta$$

(recall that  $\theta$  is the parameter vector of  $\hat{Q}_\theta$ )

# Principle

- deriving a natural actor-critic algorithm

# Principle

- deriving a natural actor-critic algorithm
  - actor update, natural gradient ascent :  $\omega_j = \omega_{j-1} + \beta_j \theta_j$

# Principle

- deriving a natural actor-critic algorithm
  - actor update, natural gradient ascent :  $\omega_j = \omega_{j-1} + \beta_j \theta_j$
  - critic update ?

# Principle

- deriving a natural actor-critic algorithm
  - actor update, natural gradient ascent :  $\omega_j = \omega_{j-1} + \beta_j \theta_j$
  - critic update ?
- $\hat{Q}_\theta$  **does not** approximate the  $Q$ -function, but the advantage function
$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s) = Q^\pi(s, a) - E_{a|s, \pi}[Q^\pi(s, a)]$$

# Principle

- deriving a natural actor-critic algorithm
  - actor update, natural gradient ascent :  $\omega_j = \omega_{j-1} + \beta_j \theta_j$
  - critic update ?
- $\hat{Q}_\theta$  **does not** approximate the  $Q$ -function, but the advantage function
 
$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s) = Q^\pi(s, a) - E_{a|s, \pi}[Q^\pi(s, a)]$$
- this is because :
 
$$E_{a|s, \pi}[\hat{Q}_\theta(s, a)] = E_{a|s, \pi}[\theta^T \nabla_\omega \ln \pi_\omega(a|s)] = 0$$

# Principle

- deriving a natural actor-critic algorithm
  - actor update, natural gradient ascent :  $\omega_j = \omega_{j-1} + \beta_j \theta_j$
  - critic update ?

- $\hat{Q}_\theta$  **does not** approximate the  $Q$ -function, but the advantage function

$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s) = Q^\pi(s, a) - E_{a|s, \pi}[Q^\pi(s, a)]$$

- this is because :

$$E_{a|s, \pi}[\hat{Q}_\theta(s, a)] = E_{a|s, \pi}[\theta^T \nabla_\omega \ln \pi_\omega(a|s)] = 0$$

- the advantage function does not satisfy a Bellman equation, deriving a critic is not straightforward

## An example : NTD

- $\hat{Q}_\theta$  is chosen such as satisfying the compatibility condition

## An example : NTD

- $\hat{Q}_\theta$  is chosen such as satisfying the compatibility condition
- as it represents the advantage function, it is not enough

## An example : NTD

- $\hat{Q}_\theta$  is chosen such as satisfying the compatibility condition
- as it represents the advantage function, it is not enough
- an approximate value function  $\hat{V}_\xi(s)$  is introduced, and learned using classical TD-learning :

$$\begin{cases} \delta_i = r_i + \gamma \hat{V}_{\xi_{i-1}}(\mathbf{s}_{i+1}) - \hat{V}_{\xi_{i-1}}(\mathbf{s}_i) \\ \xi_i = \xi_{i-1} + \alpha_i \nabla_\xi (\hat{V}_{\xi_{i-1}}(\mathbf{s}_i)) \delta_i \end{cases}$$

## An example : NTD

- $\hat{Q}_\theta$  is chosen such as satisfying the compatibility condition
- as it represents the advantage function, it is not enough
- an approximate value function  $\hat{V}_\xi(s)$  is introduced, and learned using classical TD-learning :

$$\begin{cases} \delta_i = r_i + \gamma \hat{V}_{\xi_{i-1}}(s_{i+1}) - \hat{V}_{\xi_{i-1}}(s_i) \\ \xi_i = \xi_{i-1} + \alpha_i \nabla_\xi (\hat{V}_{\xi_{i-1}}(s_i)) \delta_i \end{cases}$$

- the TD error  $\delta_i$  is used as the target for the advantage function (**bootstrap !**) :

$$\theta_i = \theta_{i-1} + \alpha_i \nabla_\xi (\hat{Q}_{\theta_{i-1}}(s_i, a_i)) (\delta_i - \hat{Q}_{\theta_{i-1}}(s_i, a_i))$$

## An example : NTD

- $\hat{Q}_\theta$  is chosen such as satisfying the compatibility condition
- as it represents the advantage function, it is not enough
- an approximate value function  $\hat{V}_\xi(s)$  is introduced, and learned using classical TD-learning :

$$\begin{cases} \delta_i = r_i + \gamma \hat{V}_{\xi_{i-1}}(s_{i+1}) - \hat{V}_{\xi_{i-1}}(s_i) \\ \xi_i = \xi_{i-1} + \alpha_i \nabla_\xi (\hat{V}_{\xi_{i-1}}(s_i)) \delta_i \end{cases}$$

- the TD error  $\delta_i$  is used as the target for the advantage function (**bootstrap !**) :

$$\theta_i = \theta_{i-1} + \alpha_i \nabla_\xi (\hat{Q}_{\theta_{i-1}}(s_i, a_i)) (\delta_i - \hat{Q}_{\theta_{i-1}}(s_i, a_i))$$

- to ensure the “good approximation condition”, the actor should seem stationary from the critic point of view :

$$\lim_{j \rightarrow \infty} \frac{\beta_j}{\alpha_j} = 0$$

- 1 Background
  - Paradigm
  - MDP and Bellman equations
  - Actor, Critic and actor-critic
- 2 Natural actor-critics
  - Policy gradient
  - Policy gradient with FA
  - Natural policy gradient with FA
  - Deriving a critic
- 3 Revisiting Natural actor-critics**
  - Semi-compatible approximation**
  - Revisiting (natural) policy gradient with FA**
  - Deriving new algorithms**
- 4 Illustration and conclusion
  - Preliminary results
  - Conclusion and perspectives

- deriving a critic is not direct...

- deriving a critic is not direct...
- Why not directly representing the  $Q$ -function ?

- deriving a critic is not direct...
- Why not directly representing the  $Q$ -function ?

### Semi-compatible approximation

$$\hat{Q}_{\theta, \xi} = f_{\theta}(s, a) + g_{\xi}(s)$$

with  $f_{\theta}(s, a)$  being compatible :

$$\nabla_{\theta} f_{\theta}(s, a) = \nabla_{\omega} \ln \pi_{\omega}(a|s)$$

- deriving a critic is not direct...
- Why not directly representing the  $Q$ -function ?

### Semi-compatible approximation

$$\hat{Q}_{\theta, \xi} = f_{\theta}(s, a) + g_{\xi}(s)$$

with  $f_{\theta}(s, a)$  being compatible :

$$\nabla_{\theta} f_{\theta}(s, a) = \nabla_{\omega} \ln \pi_{\omega}(a|s)$$

- $f_{\theta}(s, a)$  approximates the advantage function, and  $g_{\xi}(s)$  the value function

- what about theoretical results with the semi-compatibility ?

- what about theoretical results with the semi-compatibility ?
- all results presented so far still hold, with minor changes in the proofs !

- what about theoretical results with the semi-compatibility ?
- all results presented so far still hold, with minor changes in the proofs !
- this is due to a known result : the policy gradient is invariant to any state-dependent bias

$$\nabla_{\omega} \rho(\pi_{\omega}) = \sum_{s \in \mathcal{S}} d^{\pi_{\omega}}(s) \sum_{a \in \mathcal{A}} (Q^{\pi_{\omega}}(s, a) + b(s)) \nabla_{\omega} \pi_{\omega}(a|s)$$

- what about theoretical results with the semi-compatibility ?
- all results presented so far still hold, with minor changes in the proofs !
- this is due to a known result : the policy gradient is invariant to any state-dependent bias

$$\nabla_{\omega} \rho(\pi_{\omega}) = \sum_{s \in \mathcal{S}} d^{\pi_{\omega}}(s) \sum_{a \in \mathcal{A}} (Q^{\pi_{\omega}}(s, a) + b(s)) \nabla_{\omega} \pi_{\omega}(a|s)$$

- in our case,  $b(s) = g_{\xi}(s)$

# TD-NAC

- the actor is still updated according to the natural gradient ascent :

$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

# TD-NAC

- the actor is still updated according to the natural gradient ascent :
- the critic is updated according to a simple TD-learning (with the two-timescale approach) :

$$\begin{aligned}\delta_i &= r_i + \gamma \hat{Q}_{\theta_{i-1}, \xi_{i-1}}(\mathbf{s}_{i+1}, \mathbf{a}_{i+1}) - \hat{Q}_{\theta_{i-1}, \xi_{i-1}}(\mathbf{s}_i, \mathbf{a}_i) \\ \begin{pmatrix} \theta_i \\ \xi_i \end{pmatrix} &= \begin{pmatrix} \theta_{i-1} \\ \xi_{i-1} \end{pmatrix} + \alpha_i \delta_i \nabla_{\theta, \xi} \hat{Q}_{\theta_{i-1}, \xi_{i-1}}(\mathbf{s}_i, \mathbf{a}_i)\end{aligned}$$

## TD-NAC

- the actor is still updated according to the natural gradient ascent :

$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

- the critic is updated according to a simple TD-learning (with the two-timescale approach) :

$$\begin{aligned} \delta_i &= r_i + \gamma \hat{Q}_{\theta_{i-1}, \xi_{i-1}}(s_{i+1}, a_{i+1}) - \hat{Q}_{\theta_{i-1}, \xi_{i-1}}(s_i, a_i) \\ \begin{pmatrix} \theta_i \\ \xi_i \end{pmatrix} &= \begin{pmatrix} \theta_{i-1} \\ \xi_{i-1} \end{pmatrix} + \alpha_i \delta_i \nabla_{\theta, \xi} \hat{Q}_{\theta_{i-1}, \xi_{i-1}}(s_i, a_i) \end{aligned}$$

- TD-NAC is not** the same algorithm as **NTD**, however links can be drawn

# KNAC

- the actor is still updated according to the natural gradient ascent :

$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

# KNAC

- the actor is still updated according to the natural gradient ascent :

$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

- the critic is based on KTD :

$$\begin{cases} \begin{pmatrix} \theta_i \\ \xi_i \end{pmatrix} = \begin{pmatrix} \theta_{i-1} \\ \xi_{i-1} \end{pmatrix} + v_i \\ r_i = \hat{Q}_{\theta_i, \xi_i}(s_i, a_i) - \gamma \hat{Q}_{\theta_i, \xi_i}(s_{i+1}, a_{i+1}) + n_i \end{cases}$$

# KNAC

- the actor is still updated according to the natural gradient ascent :

$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

- the critic is based on KTD :

$$\begin{cases} \begin{pmatrix} \theta_i \\ \xi_i \end{pmatrix} = \begin{pmatrix} \theta_{i-1} \\ \xi_{i-1} \end{pmatrix} + v_i \\ r_i = \hat{Q}_{\theta_i, \xi_i}(s_i, a_i) - \gamma \hat{Q}_{\theta_i, \xi_i}(s_{i+1}, a_{i+1}) + n_i \end{cases}$$

- two main ideas :

# KNAC

- the actor is still updated according to the natural gradient ascent :

$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

- the critic is based on KTD :

$$\begin{cases} \begin{pmatrix} \theta_i \\ \xi_i \end{pmatrix} = \begin{pmatrix} \theta_{i-1} \\ \xi_{i-1} \end{pmatrix} + v_i \\ r_i = \hat{Q}_{\theta_i, \xi_i}(s_i, a_i) - \gamma \hat{Q}_{\theta_i, \xi_i}(s_{i+1}, a_{i+1}) + n_i \end{cases}$$

- two main ideas :
  - use a second order critic instead of a first order one  $\implies$  sample efficiency

# KNAC

- the actor is still updated according to the natural gradient ascent :

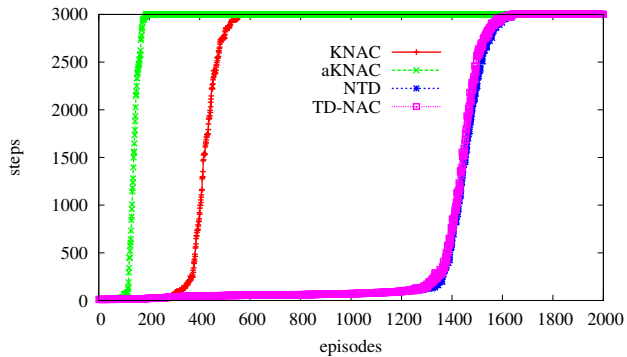
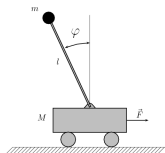
$$\omega_i = \omega_{i-1} + \beta_i \theta_i$$

- the critic is based on KTD :

$$\begin{cases} \begin{pmatrix} \theta_i \\ \xi_i \end{pmatrix} = \begin{pmatrix} \theta_{i-1} \\ \xi_{i-1} \end{pmatrix} + v_i \\ r_i = \hat{Q}_{\theta_i, \xi_i}(s_i, a_i) - \gamma \hat{Q}_{\theta_i, \xi_i}(s_{i+1}, a_{i+1}) + n_i \end{cases}$$

- two main ideas :**
  - use a second order critic instead of a first order one  $\implies$  sample efficiency
  - use an adaptive critic (through noise  $v_i$ ) instead of the two time-scale approach (actor stationarity  $\rightarrow$  “good approximation” condition)

- 1 Background
  - Paradigm
  - MDP and Bellman equations
  - Actor, Critic and actor-critic
- 2 Natural actor-critics
  - Policy gradient
  - Policy gradient with FA
  - Natural policy gradient with FA
  - Deriving a critic
- 3 Revisiting Natural actor-critics
  - Semi-compatible approximation
  - Revisiting (natural) policy gradient with FA
  - Deriving new algorithms
- 4 **Illustration and conclusion**
  - **Preliminary results**
  - **Conclusion and perspectives**



- contributions :
  - slight variation of existing theoretical results allowing to work directly with the state-action value function
  - some of possible resulting algorithms
  - the idea of using an adaptive critic

- contributions :
  - slight variation of existing theoretical results allowing to work directly with the state-action value function
  - some of possible resulting algorithms
  - the idea of using an adaptive critic
- perspectives :
  - new critics
  - theoretical insights into the adaptive critic aspect
  - more flexible representation than the semi-compatibility condition ?