Policy Gradient Approaches for Multi-Objective Sequential Decision Making

Simone Parisi, Matteo Pirotta, Nicola Smacchia
Luca Bascetta, Marcello Restelli

Department of Electronic, Information and Bioengineering
Politecnico di Milano, Italy

IEEE WCCI Conference
International Joint Conference on Neural Networks, 2014
Beijing, China

July 6-11, 2014

Outline

Motivations and Goals

GOALS:

- Analysis of advantages and limits of MOMDP techniques
- Explore **Policy Gradient** in MOMDPS

MOTIVATIONS:

- Policy Gradient techniques are widespread in RL
- Real–world applications are often multi–objectives
- Literature provides few MORL algorithms

Outline

Contributes

ALGORITHMIC: we propose two MORL policy gradient algorithms

- Radial Algorithm (RA)
- Pareto Following Algorithm (PFA)

EMPIRICAL: several test have been performed on different domains in order to evaluate proposed algorithms

- Linear–Quadratic–Gaussian regulator
- Deep Sea Treasure
- Water Reservoir

ANALYTIC:

- As far as we now, it is the first deep analysis of MORL Policy Gradient algorithms after Shelton (2001)
- and the first analysis on the use of **stochastic policies** in MORL

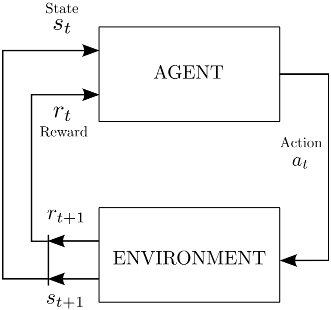
Outline

Markov Decision Process (MDP)

MDP:

$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma, D \rangle$$

$$J(\pi) = \mathbb{E}\left[\sum_{t=1}^{T} \gamma^{t-1} r_t | \pi, s_0 \sim D\right]$$
$$= \int_{s \in \mathcal{S}} d^{\pi}(s) \int_{a \in \mathcal{A}} \pi(a|s) \mathcal{R}(s,a) \mathrm{d}a\mathrm{d}s$$



MDP algorithms

- Dynamic Programming
- Linear Programming
- **Reinforcement Learning**

Algorithms for continuous MDPs

- **Policy Gradient**
- Genetic Algorithms
- Classification–based algorithms

MDP: Trajectory–based Policy Gradient

Parametric policy model: $\pi(a|s,\boldsymbol{\theta})$, e.g., **Gauss** or **Gibbs** policy models

Gradient Estimate:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \mathbb{E}_{\tau \sim p(\cdot|\boldsymbol{\theta})} \left[\nabla_{\boldsymbol{\theta}} \log p(\tau|\boldsymbol{\theta}) R(\tau) \right]$$

$$R(\tau) = \sum_{t=1}^{T} \gamma^{t-1} r_t$$

Gradient Ascent:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha_t \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

Advantages

- Continuous state and action space
- On–policy and "off-policy" learning
- Direct learning in the policy space

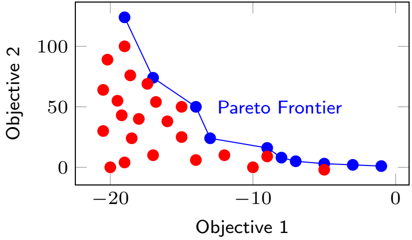
Drawbacks

- Local Optimum
- High–variance gradient estimate
- Tuning of the learning step α_t

Multi–Objective MDPs

VECTORIAL RETURN

$$\boldsymbol{J}(\pi) = [J_1(\pi), J_2(\pi), \dots, J_q(\pi)]^{\mathsf{T}}$$
$$= \mathbb{E}\left[\sum_{t=1}^{T} \gamma^{t-1} \mathbf{r}_t | \pi, s_0 \sim D\right]$$



SOLUTION CONCEPT: Pareto Dominance

$$\boldsymbol{J}(\pi) \succ \boldsymbol{J}(\bar{\pi}) \Leftrightarrow (\exists k \mid J_k(\pi) > J_k(\bar{\pi})) \wedge (\nexists h \mid J_h(\pi) < J_h(\bar{\pi})).$$

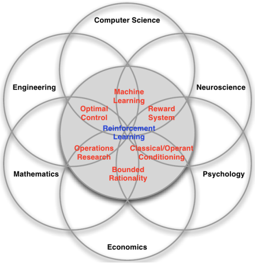
PARETO FRONTIER in policy space

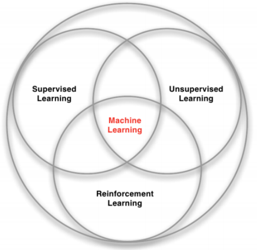
$$\Pi^* = \{\pi^* \in \Pi \;:\; \nexists \pi \in \Pi \mid \boldsymbol{J}(\pi) \succ \boldsymbol{J}(\pi^*)\}$$

Multi–Objective MDPs – 2

Solve MOMDPs:

- Reinforcement Learning
  - Single–policy vs Multiple–policies
  - Linear scalarization vs Non-linear scalarization
- Mathematical Optimization
  - Evolutionary Algorithms
  - Gradient Ascent





Multi–Objective Policy Gradient

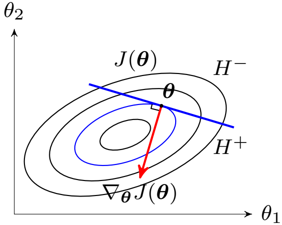
Concepts

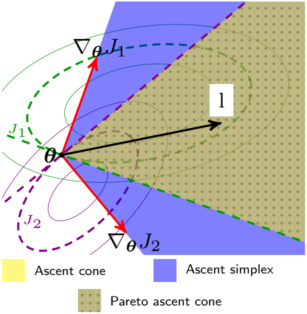
- Half Spaces
- Ascent Cone

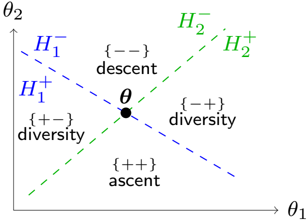
  $C(\boldsymbol{\theta}) = \{\boldsymbol{l} : \boldsymbol{l} \cdot \nabla_{\boldsymbol{\theta}} J_i(\boldsymbol{\theta}) \geq 0\}$
- Ascent Simplex

  $S(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \sum_{i=1}^{q} \lambda_i \nabla_{\boldsymbol{\theta}} J_i(\boldsymbol{\theta})$
- **Pareto–Ascent Cone**

  $S(\boldsymbol{\lambda}, \boldsymbol{\theta}) \cap C(\boldsymbol{\theta})$





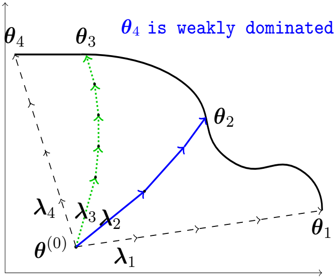


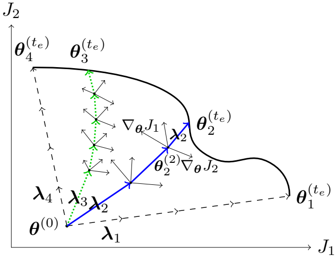
Null Pareto–Ascent Cone ⇒ (local) optimal solution

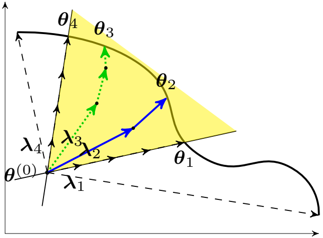
Radial Algorithm

Idea: p gradient ascent searches are performed, each one following a different, *uniformly spaced* direction in the **ascent simplex**

Problem: weak optimal solutions





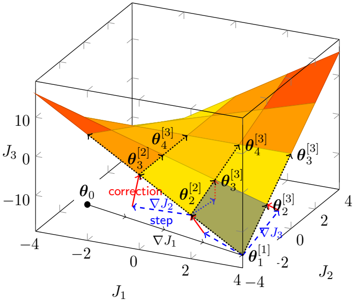


Pareto Following Algorithm

Phase 1: A solution on the Pareto frontier is reached by considering a single objective

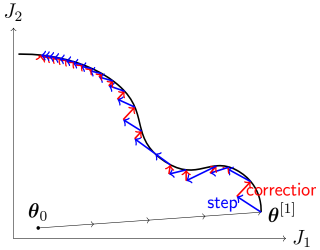
Phase 2: Exploration

- Improvement step: move the solution toward **one** objective at a time
- Correction step: improvement may lead the point outside the frontier. Correction **moves** the point again **on the frontier**



Problems:

- Can reach deterministic policies
- Need to **reintroduce stochasticity** (e.g., based on the entropy)
- Tuning of learning rate



Outline

Experiments

Multi–Objective learning difficulties

- More than two objectives
- **Continuous** state and action space
- **Stochastic** environments
- **Concave** Pareto frontiers

Domains

- Linear–Quadratic–Gaussian regulator
- `Deep Sea Treasure`
- `Water Reservoir`

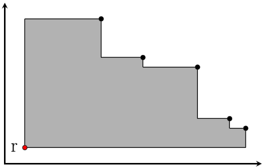
Evaluation Criteria

Loss – Castelletti et al. (2013)

$$l(\widehat{\mathcal{J}}^M, \mathcal{J}^*, W, p) = \int_{w \in W} \frac{J_w^* - \widehat{J}_w^{M,*}}{\Delta J_w^*} p(\mathrm{d}w)$$

$$\Delta J_w^* = \sum_{i=1}^{q} w_i \left(\max_{\bar{w} \in W} J_{\bar{w},i}^* - \min_{\bar{w} \in W} J_{\bar{w},i}^* \right)$$

Hyper volume – Zitzler et al. (2003)



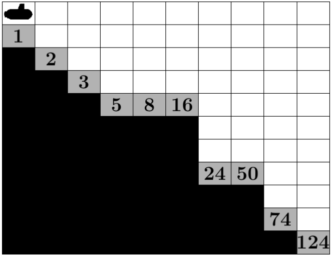
Comparison Algorithms

- Multi–Objective Fitted Q–iteration (MOFQI) (Castelletti et al., 2012)
- S–Metric Selection Evolutionary Multi–Objective Algorithm (SMS-EMOA) (Beume et al., 2007)

Deep Sea Treasure (Vamplew et al., 2008)

GOALS: `Vamplew et al. (2008)`

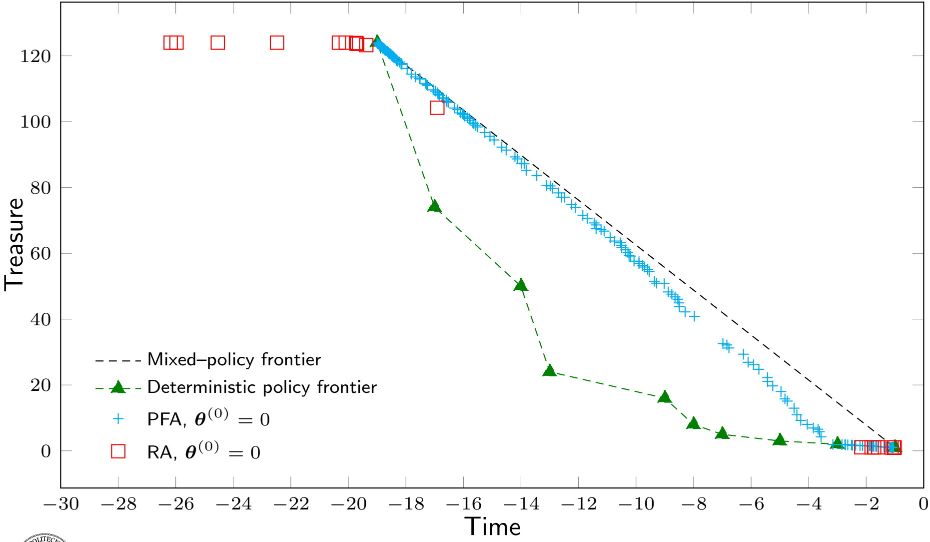
- Maximise treasure value
- Minimise time steps



FEATURES:

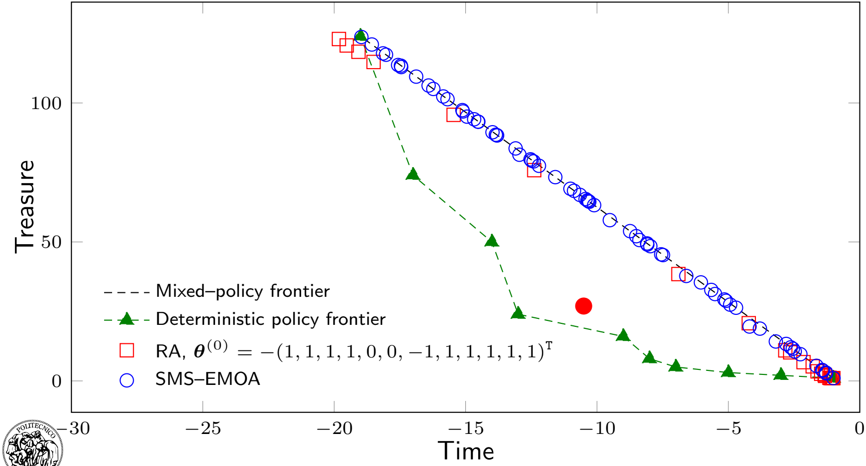
- Deterministic Pareto frontier is concave
- Optimal Frontier is obtained by **mixture–policy** that can be approximate by stochastic policies

Deep Sea Treasure – 2



Deep Sea Treasure – 3

Algorithm	Hyper–Volume	# Policies
PFA	0.4589	2,012
RA	0.3999	2,256
SMS–EMOA	0.4895	6,200



Water Reservoir (Castelletti et al., 2013)

MODEL: `Castelletti et al. (2013)`

$$s_{t+1} = s_t + \epsilon_{t+1} - \max(\underline{a}_t, \min(\bar{a}_t, u_t))$$

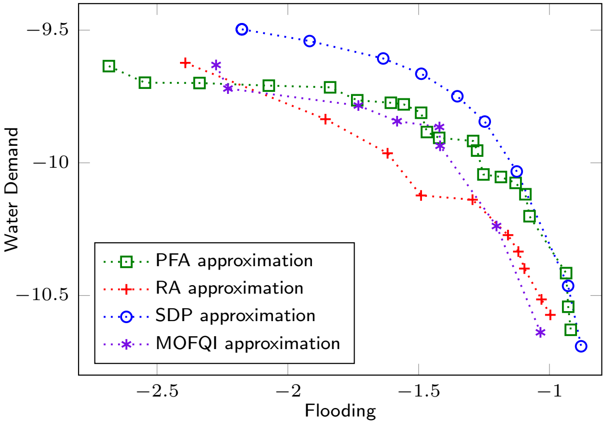
RESERVOIR INFLOW

$$e_{t+1} = \mathcal{N}(40, 100)$$

REWARD FUNCTIONS

$$\mathcal{R}_1(s_t, a_t, s_{t+1}) = -\max(h_{t+1} - \bar{h}, 0) \qquad \textit{flooding}$$
$$\mathcal{R}_2(s_t, a_t, s_{t+1}) = -\max(\bar{\rho} - \rho_t, 0) \qquad \textit{irrigation supply}$$
$$\mathcal{R}_3(s_t, a_t, s_{t+1}) = -\max(\bar{e}_t - e_{t+1}, 0) \qquad \textit{hydropower supply}$$

Water Reservoir – 2



Algorithm	Loss (2-obj.)	Loss (3-obj.)
Radial	0.0945	0.0123
Pareto following	0.0811	0.0162
MOFQI (Pianosi et al., 2013)	0.1870	0.0540
FQI (Ernst et al., 2005)	0.1910	0.0292

Outline

Conclusions

Advantages

Policy Gradient

- **Continuous** state and action space
- **Arbitrary** number of objectives
- **Stochastic** policies
- Works with **concave** Pareto frontiers

PFA and RA features

- **Scalable**
- **Parallel**

Drawbacks

RA:

- Lost of performances when there are weakly dominated solutions

PFA:

- Require randomization
- Highly sensible to learning parameters

Future Works

Algorithm related works

- **Radial Algorithm**
  - Different sample methods in order to avoid weakly dominated solutions
- **Pareto Path Following Algorithm**
  - Investigate policy randomization

Other branches

- Pareto Frontier Functional approximation `(Pirotta et al., 2014)`
- Off–Policy algorithms

Thank you for your attention

Slides and source code at:

http://home.dei.polimi.it/pirotta

References

Beume, N., Naujoks, B., and Emmerich, M. (2007). Sms-emoa: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669.

Castelletti, A., Pianosi, F., and Restelli, M. (2012). Tree-based fitted q-iteration for multi-objective markov decision problems. In *IJCNN*, pages 1–8. IEEE.

Castelletti, A., Pianosi, F., and Restelli, M. (2013). A multiobjective reinforcement learning approach to water resources systems operation: Pareto frontier approximation in a single run. *Water Resources Research*, 49(6):3476–3486.

Ernst, D., Geurts, P., and Wehenkel, L. (2005). Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556.

Pianosi, F., Castelletti, A., and Restelli, M. (2013). Tree-based fitted q-iteration for multi-objective markov decision processes in water resource management. *Journal of Hydroinformatics*, 15(2):258–270.

Pirotta, M., Parisi, S., and Restelli, M. (2014). Multi–objective Reinforcement Learning with Continuous Pareto Frontier Approximation. *arXiv:1406.3497*.

Shelton, C. R. (2001). *Importance Sampling for Reinforcement Learning with Multiple Objectives*. PhD thesis, Massachusetts Institute of Technology.

Vamplew, P., Yearwood, J., Dazeley, R., and Berry, A. (2008). On the limitations of scalarisation for multi-objective reinforcement learning of pareto fronts. In *Advances in Artificial Intelligence*, pages 372–378. Springer.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., and da Fonseca, V. (2003). Performance assessment of multiobjective optimizers: an analysis and review. *IEEE T EVOLUT COMPUT*, 7(2):117–132.

LQG

Single Objective

$$s_{t+1} = A\, s_t + B\, a_t, \qquad a_t \sim \mathcal{N}(K \cdot s_t, \Sigma)$$

$$r_t = -s_t^{\mathsf{T}} Q s_t - a_t^{\mathsf{T}} R a_t$$

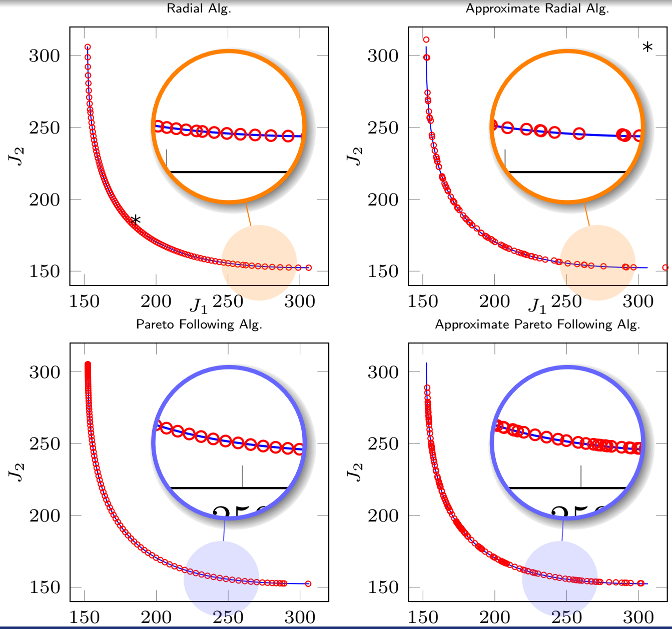
Multi Objective

$$\mathcal{R}_i(s,a) = -s_i^2 - \sum_{i \neq j} a_j^2$$

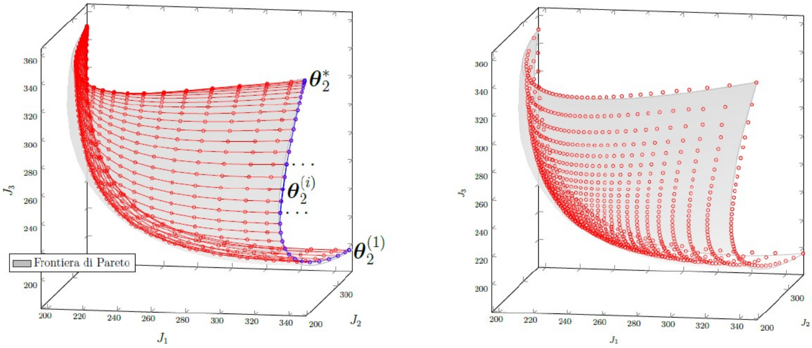
Why it is interesting?

- The Pareto frontier is known
- Closed–form for performance measure $\mathbf{J}$
- It is possible to evaluate algorithm behaviours in exact and approximate scenarios

LQG – 2


Radial Alg.
Approximate Radial Alg.
Pareto Following Alg.
Approximate Pareto Following Alg.
J_2
J_1
300
250
200
150
150
200
250
300

LQG – 3



Algorithm	α_s	α_c	ϵ_s	ϵ_c	Directions	Iterations	Time	Solutions	Loss
PFA	0.005	0.1	0.01	0.01	-	2,317	21s	943	$3.34e-04$
RA	0.5	-	0.01	-	989	3,840	17s	989	$9.45e-05$