Deep Deterministic Policy Gradient - Derivations & Proofs

Paul F. Roysdon, Ph.D.

Contents

1 Mathematical Derivations & Proofs

1.1 Introduction

Deep Deterministic Policy Gradient (DDPG) is an *off-policy* actor–critic algorithm for continuous-action control. It combines (i) the *Deterministic Policy Gradient* (DPG) theorem, which yields a tractable gradient for deterministic policies, with (ii) a value function (critic) trained by *Bellman residual minimization*, and (iii) two stabilizing devices: *experience replay* and *target networks* with *soft updates*. We derive DDPG from first principles: starting at the Markov Decision Process (MDP) objective, proving the DPG theorem (proof sketch), deriving the critic Bellman target and loss, and presenting the resulting stochastic-gradient updates.

1.2 Data and Notation

Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$ be a discounted MDP with state space $\mathcal{S}$, *continuous* action space $\mathcal{A} \subseteq \mathbb{R}^m$, transition kernel $P(\mathbf{s}' \mid \mathbf{s}, \mathbf{a})$, reward $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and discount $\gamma \in [0, 1)$. A deterministic policy $\mu_{\boldsymbol{\theta}} : \mathcal{S} \to \mathcal{A}$ is parameterized by $\boldsymbol{\theta} \in \mathbb{R}^p$; a critic (action-value) is $Q_{\boldsymbol{\phi}} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ with $\boldsymbol{\phi} \in \mathbb{R}^q$. Denote the discounted state visitation measure

$$\rho^{\mu}(\mathbf{s}) \triangleq (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \Pr(\mathbf{s}_t = \mathbf{s} \mid \mathbf{s}_0 \sim \rho_0,\ \mathbf{a}_k = \mu_{\boldsymbol{\theta}}(\mathbf{s}_k)).$$

Dimensions and properties: $\boldsymbol{\theta} \in \mathbb{R}^p$ (column), $\boldsymbol{\phi} \in \mathbb{R}^q$; actions $\mathbf{a} = \mu_{\boldsymbol{\theta}}(\mathbf{s}) \in \mathbb{R}^m$; Jacobian $\nabla_{\boldsymbol{\theta}} \mu_{\boldsymbol{\theta}}(\mathbf{s}) \in \mathbb{R}^{p \times m}$; action-value gradient $\nabla_{\mathbf{a}} Q_{\boldsymbol{\phi}}(\mathbf{s}, \mathbf{a}) \in \mathbb{R}^m$.

1.3 Objective and Value Functions

The control objective is the expected discounted return under $\mu_{\boldsymbol{\theta}}$:

$$J(\boldsymbol{\theta}) \;=\; \mathbb{E}_{\mathbf{s}_0\sim\rho_0}\left[\sum_{t=0}^{\infty}\gamma^t\, r\big(\mathbf{s}_t,\mu_{\boldsymbol{\theta}}(\mathbf{s}_t)\big)\right] \;=\; \frac{1}{1-\gamma}\,\mathbb{E}_{s\sim\rho^{\mu}}\big[\,r\big(\mathbf{s},\mu_{\boldsymbol{\theta}}(\mathbf{s})\big)\big]\,. \tag{1}$$

The action-value under μ is

$$Q^{\mu}(\mathbf{s},\mathbf{a}) \;=\; \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t r(\mathbf{s}_t,\mathbf{a}_t)\;\middle|\;\mathbf{s}_0=\mathbf{s},\;\mathbf{a}_0=\mathbf{a},\;\mathbf{a}_{t\geq 1}=\mu(\mathbf{s}_t)\right], \tag{2}$$

and satisfies the Bellman equation

$$Q^{\mu}(\mathbf{s},\mathbf{a}) \;=\; r(\mathbf{s},\mathbf{a})+\gamma\,\mathbb{E}_{\mathbf{s}'\sim P(\cdot|\mathbf{s},\mathbf{a})}\big[\,Q^{\mu}\big(\mathbf{s}',\mu(\mathbf{s}')\big)\big]\,. \tag{3}$$

1.4 Deterministic Policy Gradient (DPG) Theorem

Statement. If $\mu_{\boldsymbol{\theta}}$ is deterministic and differentiable in $\boldsymbol{\theta}$, then

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \;=\; \frac{1}{1-\gamma}\,\mathbb{E}_{\mathbf{s}\sim\rho^{\mu}}\Big[\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(\mathbf{s})\;\nabla_{\mathbf{a}}Q^{\mu}(\mathbf{s},\mathbf{a})\big|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(\mathbf{s})}\Big]\,. \tag{4}$$

Moreover, this gradient is *off-policy*: the expectation may be taken with respect to any state distribution ρ^{β} induced by a behavior policy β that covers the support of ρ^{μ}, without biasing $\nabla_{\boldsymbol{\theta}}J(\boldsymbol{\theta})$.

Proof. Define the state-value $V^{\mu}(\mathbf{s})=Q^{\mu}\big(\mathbf{s},\mu(\mathbf{s})\big)$. Differentiate $J(\boldsymbol{\theta})=(1-\gamma)^{-1}\,\mathbb{E}_{\mathbf{s}\sim\rho^{\mu}}[V^{\mu}(\mathbf{s})]$:

$$\nabla_{\boldsymbol{\theta}}J=\frac{1}{1-\gamma}\int\big(\nabla_{\boldsymbol{\theta}}\rho^{\mu}(\mathbf{s})\big)\,V^{\mu}(\mathbf{s})\,d\mathbf{s}+\frac{1}{1-\gamma}\int\rho^{\mu}(\mathbf{s})\,\nabla_{\boldsymbol{\theta}}V^{\mu}(\mathbf{s})\,d\mathbf{s}.$$

A sensitivity lemma (see Silver et al., 2014) shows the first term cancels with part of the second by the stationarity of ρ^{μ} and the Bellman equation for V^{μ}; what remains depends only on the *direct* effect of $\boldsymbol{\theta}$ on actions through μ:

$$\nabla_{\boldsymbol{\theta}}V^{\mu}(\mathbf{s})=\nabla_{\boldsymbol{\theta}}\mu_{\boldsymbol{\theta}}(\mathbf{s})\;\nabla_{\mathbf{a}}Q^{\mu}(\mathbf{s},\mathbf{a})\big|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(\mathbf{s})}.$$

Substitute and obtain Eqn. (4). Off-policy validity follows since the integrand does not depend on the *sampling* distribution (only on ρ^{μ} in the true gradient), allowing unbiased estimation from any ρ^{β} that sufficiently explores. ■

1.5 Critic Learning via Bellman Residual Minimization

We approximate Q^{μ} by $Q_{\boldsymbol{\phi}}$ and fit $\boldsymbol{\phi}$ by minimizing the MSE of one-step Bellman targets. Introduce *target* networks $Q_{\boldsymbol{\phi}^-}$, $\mu_{\boldsymbol{\theta}^-}$ (parameters $\boldsymbol{\phi}^-,\boldsymbol{\theta}^-$ are held fixed during target computation). For a transition $(\mathbf{s},\mathbf{a},r,\mathbf{s}')$, define

$$y \;=\; r+\gamma\,Q_{\boldsymbol{\phi}^-}\big(\mathbf{s}',\,\mu_{\boldsymbol{\theta}^-}(\mathbf{s}')\big). \tag{5}$$

The critic loss over a minibatch $\mathcal{B}$ is

$$\mathcal{L}(\boldsymbol{\phi}) \;=\; \frac{1}{|\mathcal{B}|}\sum_{(\mathbf{s},\mathbf{a},r,\mathbf{s}')\in\mathcal{B}}\big(Q_{\boldsymbol{\phi}}(\mathbf{s},\mathbf{a})\;-\;y\big)^2\,. \tag{6}$$

Contraction argument. For fixed μ and $\gamma<1$, the Bellman operator $\mathcal{T}^{\mu}[Q](\mathbf{s},\mathbf{a})=r(\mathbf{s},\mathbf{a})+\gamma\,\mathbb{E}[Q(\mathbf{s}',\mu(\mathbf{s}'))]$ is a γ-contraction in $\|\cdot\|_{\infty}$ with unique fixed point Q^{μ}. Minimizing Eqn. (6) (with sufficiently expressive $Q_{\boldsymbol{\phi}}$ and small step sizes) tracks this fixed point.

1.6 Actor Update (Chain Rule Form of DPG)

Using the learned critic as a surrogate for Q^μ, the DPG update is the sample average of the chain rule term:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \;\approx\; \frac{1}{|\mathcal{B}|} \sum_{s \in \mathcal{B}_{\mathbf{s}}} \Big[\nabla_{\boldsymbol{\theta}} \mu_{\boldsymbol{\theta}}(\mathbf{s})\, \nabla_{\mathbf{a}} Q_{\boldsymbol{\phi}}(\mathbf{s}, \mathbf{a})\big|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(\mathbf{s})} \Big], \qquad \boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha_{\text{act}}\, \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}). \tag{7}$$

Dimensions. Each summand is in $\mathbb{R}^p$: $\nabla_{\boldsymbol{\theta}} \mu_{\boldsymbol{\theta}}(\mathbf{s}) \in \mathbb{R}^{p \times m}$ times $\nabla_{\mathbf{a}} Q_{\boldsymbol{\phi}}(\mathbf{s}, \mathbf{a}) \in \mathbb{R}^m$.

1.7 Experience Replay and Target Networks

Replay buffer $\mathcal{D}$. Transitions $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$ are stored in a buffer $\mathcal{D}$; minibatches $\mathcal{B} \subset \mathcal{D}$ are sampled i.i.d., reducing the temporal correlation of updates and enabling off-policy learning. Exploration is injected *only at data collection time* via a behavior policy

$$\mathbf{a}_t \;=\; \mu_{\boldsymbol{\theta}}(\mathbf{s}_t) + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2 I) \;\text{ or OU noise.}$$

Soft target updates. After each gradient step, update target parameters by *Polyak averaging*:

$$\boldsymbol{\theta}^- \leftarrow \tau\, \boldsymbol{\theta} + (1-\tau)\, \boldsymbol{\theta}^-, \qquad \boldsymbol{\phi}^- \leftarrow \tau\, \boldsymbol{\phi} + (1-\tau)\, \boldsymbol{\phi}^-, \quad \text{with } \tau \in (0, 1]. \tag{8}$$

This stabilizes the bootstrapped target Eqn. (5).

1.8 DDPG Algorithm (One Iteration)

1. **Collect data.** Execute behavior $\mathbf{a}_t = \mu_{\boldsymbol{\theta}}(\mathbf{s}_t) + \varepsilon_t$, observe $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$, append to $\mathcal{D}$.
2. **Sample minibatch.** Draw $\mathcal{B} = \{(\mathbf{s}_i, \mathbf{a}_i, r_i, \mathbf{s}'_i)\}_{i=1}^N \sim \mathcal{D}$.
3. **Critic target.** For each i, compute $y_i = r_i + \gamma\, Q_{\boldsymbol{\phi}^-}(\mathbf{s}'_i, \mu_{\boldsymbol{\theta}^-}(\mathbf{s}'_i))$.
4. **Critic update.** Descend the loss $\mathcal{L}(\boldsymbol{\phi}) = \frac{1}{N} \sum_i \big(Q_{\boldsymbol{\phi}}(\mathbf{s}_i, \mathbf{a}_i) - y_i\big)^2$: $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \alpha_{\text{crt}} \nabla_{\boldsymbol{\phi}} \mathcal{L}$.
5. **Actor update.** Ascend Eqn. (7) with states $\{\mathbf{s}_i\}$: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha_{\text{act}} \frac{1}{N} \sum_i \nabla_{\boldsymbol{\theta}} \mu_{\boldsymbol{\theta}}(\mathbf{s}_i)\, \nabla_{\mathbf{a}} Q_{\boldsymbol{\phi}}(\mathbf{s}_i, \mathbf{a})\big|_{\mathbf{a}=\mu_{\boldsymbol{\theta}}(\mathbf{s}_i)}$.
6. **Targets.** Soft-update $(\boldsymbol{\theta}^-, \boldsymbol{\phi}^-)$ by Eqn. (8).

1.9 Practical Considerations

- **Terminal handling.** If $\mathbf{s}'$ is terminal, set the target $y = r$ (or equivalently $\gamma = 0$).
- **Normalization / clipping.** Normalize states and reward; clip gradients to control exploding updates.
- **Regularization.** Add L2 weight decay to the critic; optionally add action *bounds* by squashing $\mu_{\boldsymbol{\theta}}(\mathbf{s})$ (e.g. tanh) and rescaling to $\mathcal{A}$.
- **Entropy/exploration.** Exploration is extrinsic noise on actions since the policy is deterministic; anneal σ over training.

1.10 Connections and Guarantees

- **Bellman contraction.** For fixed μ, $\mathcal{T}^\mu$ is a γ-contraction with unique fixed point Q^μ; hence critic targets are well-posed.
- **Off-policy gradient.** The DPG gradient integrand does not depend on the sampling policy, enabling unbiased estimation from replay (behavior β), provided coverage conditions hold.
- **Actor–critic consistency.** If $Q_{\boldsymbol{\phi}} \to Q^\mu$ and $\nabla_{\boldsymbol{\theta}} \mu$ is accurately computed, the actor update Eqn. (7) is a stochastic ascent step on $J(\boldsymbol{\theta})$.

1.11 Summary of Variables and Their Dimensions

- $\boldsymbol{\theta} \in \mathbb{R}^p$: actor parameters (vector). $\mu_{\boldsymbol{\theta}} : \mathcal{S} \to \mathbb{R}^m$.
- $\boldsymbol{\phi} \in \mathbb{R}^q$: critic parameters (vector). $Q_{\boldsymbol{\phi}} : \mathcal{S} \times \mathbb{R}^m \to \mathbb{R}$.
- $\rho^\mu : \mathcal{S} \to \mathbb{R}_{\geq 0}$: discounted state visitation measure.
- $\nabla_{\boldsymbol{\theta}} \mu_{\boldsymbol{\theta}}(s) \in \mathbb{R}^{p \times m}$: actor Jacobian. $\nabla_a Q_{\boldsymbol{\phi}}(s, a) \in \mathbb{R}^m$: action gradient.
- $\alpha_{\text{act}}, \alpha_{\text{crt}} > 0$: actor/critic step sizes; $\tau \in (0, 1]$: target update rate.
- $\mathcal{D}$: replay buffer; $\mathcal{B}$: minibatch; $N = |\mathcal{B}|$.

1.12 Summary

DDPG arises by (i) adopting a deterministic, differentiable policy $\mu_{\boldsymbol{\theta}}$; (ii) applying the DPG theorem to obtain the chain-rule policy gradient Eqn. (4); (iii) learning a critic by Bellman residual minimization Eqn. (6) with stabilized targets Eqn. (5); and (iv) using replay and target networks to decorrelate updates and tame bootstrapping. The resulting actor step Eqn. (7) performs stochastic ascent on $J(\boldsymbol{\theta})$ using the critic as a surrogate for Q^μ, enabling sample-efficient control in continuous action spaces.