SUMMARY AND IMPELEMENTATION OF POLICY GRADIENT FOR THE LINEAR QUADRATIC REGULATOR

Xiangyu Liu
Shanghai Jiao Tong University
liuxiangyu999@sjtu.edu.cn

ABSTRACT

In this technical report, we will first review and summarize the paper which applies policy gradient methods directly to the linear quadratic regulator (LQR) problem. Its main results lie in two parts. (1) It proves methods with oracle access to the exact gradients enjoy global convergence. (2) Simulation-based, model-free policy gradient methods also lead to globally optimal policies, with both polynomial computational and sample complexities. We implement the model-free algorithm provided by this paper and verify this algorithm is truly effective.

1 PROBLEM BACKGROUD

This paper mainly studies the linear quadratic regulator (LQR) problem, which can be described as

$$\begin{aligned} \text{minimize} \quad & \mathbb{E}\left[\sum_{t=0}^{\infty}\left(x_t^\top Q x_t + u_t^\top R u_t\right)\right] \\ \text{such that} \quad & x_{t+1} = Ax_t + Bu_t, \quad x_0 \sim \mathcal{D} \end{aligned}$$

where initial state $x_0 \sim \mathcal{D}$ is assumed to be randomly distributed according to distribution $\mathcal{D}$; the matrices $A \in \mathbb{R}^{d\times d}$ and $B \in \mathbb{R}^{d\times k}$ are referred to as system (or transition) matrices; $Q \in \mathbb{R}^{d\times d}$ and $R \in \mathbb{R}^{k\times k}$ are both positive definite matrices that parameterize the quadratic costs.

Optimal control theory (Anderson & Moore, 2007) shows that the optimal control input can be written as a linear function in the state[1]

$$u_t = K^* x_t$$

where K^* is given as:

$$K^* = -\left(B^T P B + R\right)^{-1} B^T P A$$

P is a positive definite matrix, which is given by solving the Algebraic Riccati Equation (ARE),

$$P = A^T P A + Q - A^T P B \left(B^T P B + R\right)^{-1} B^T P A$$

2 POLICY GRADIENT METHODS FOR LQR

In this part we will introduce the exact gradient methods for the LQR problem. Since the optimal control input can be written as a linear function in the state, we can directly parameterize the optimal input by:

$$u_t = -K x_t$$

for $t \geq 0$. The cost of this K is denoted as

$$C(K) := \mathbb{E}_{x_0\sim\mathcal{D}}\left[\sum_{t=0}^{\infty}\left(x_t^\top Q x_t + u_t^\top R u_t\right)\right]$$

[1] I think the original paper made a small mistake on the optimal input. According to its methods of calculating P and K, u_t should be equal to $K^* x_t$ instead of $-K^* x_t$. It can be verified by a simple case, where $A, B, Q = 1$ and $R = 0$. Proof can be also found from Wikipedia

where $\{x_t, u_t\}$ is the trajectory induced by following K.

Gradient descent on $C(K)$, with a fixed stepsize η, follows the update rule:

$$K \leftarrow K - \eta \nabla C(K).$$

In order to do optimization, we need the exact form of this gradient. Define P_K as the solution to:

$$P_K = Q + K^\top R K + (A - BK)^\top P_K (A - BK).$$

and, under this definition, it follows that $C(K)$ can be written as:

$$C(K) = \mathbb{E}_{x_0 \sim \mathcal{D}} x_0^\top P_K x_0.$$

Also, define Σ_K as the (un-normalized) state correlation matrix, i.e.

$$\Sigma_K = \mathbb{E}_{x_0 \sim \mathcal{D}} \sum_{t=0}^{\infty} x_t x_t^\top.$$

Lemma 1 *(Policy Gradient Expression) The policy gradient is:*

$$\nabla C(K) = 2\left(\left(R + B^\top P_K B\right) K - B^\top P_K A\right) \Sigma_K$$

For simplicity, define E_K to be

$$E_K = \left(\left(R + B^\top P_K B\right) K - B^\top P_K A\right),$$

as a result the gradient can be written as $\nabla C(K) = 2 E_K \Sigma_K$

This lemma can be simply proved by using recursion and the transion equation: $x_1 = (A - BK)x_0$

3 Main Results

Results of this paper mainly lie in two parts. *The starting point is analysis on exact gradient methods. Based on the model-based case, this paper further explore theoretical properties of the model-free case which is approached by zero-order optimization.*

3.1 Model-based optimization

This paper analyze three kinds of exact update rules, gradient descent, natural policy gradient descent and Gauss-Newton method. As expected, the more complex oracle the update rule requires, the stronger convergence rate it will guarantee. All of the three update rules enjoy global convergence which is described by the following thereom:

Theorem 1 *(Global Convergence of Gradient Methods) Define $\mu := \sigma_{\min}\left(\mathbb{E}_{x_0 \sim \mathcal{D}} x_0 x_0^\top\right)$. Suppose $C(K_0)$ is finite and $\mu > 0$. Given ϵ, we want to achieve following performance bound:*

$$C(K_N) - C(K^*) \leq \varepsilon$$

Then with an appropriate step size η, iterations the algorithm needs will be:

$$N \geq \frac{\|\Sigma_{K^*}\|}{\mu} \log \frac{C(K_0) - C(K^*)}{\varepsilon}$$

$$N \geq \frac{\|\Sigma_{K^*}\|}{\mu} \left(\frac{\|R\|}{\sigma_{\min}(R)} + \frac{\|B\|^2 C(K_0)}{\mu \sigma_{\min}(R)}\right) \log \frac{C(K_0) - C(K^*)}{\varepsilon}$$

$$N \geq \frac{\|\Sigma_{K^*}\|}{\mu} \log \frac{C(K_0) - C(K^*)}{\varepsilon} \operatorname{poly}\left(\frac{C(K_0)}{\mu \sigma_{\min}(Q)}, \|A\|, \|B\|, \|R\|, \frac{1}{\sigma_{\min}(R)}\right)$$

for Gauss-Newton case, Natural policy gradient and Gradient descent respectively.

This theorem points out that exact gradient methods can be quite efficient. Whichever oracle we access to, total iterations we need is $\widetilde{\mathcal{O}}(\log \frac{1}{\varepsilon})$.

3.2 MODEL-FREE OPTIMIZATION

Based on the model-based case, the paper further analyzes settings where the controller has only simulation access to the model. *The model-free algorithm is quite intuitive and the main idea is to estimate the exact gradient by samples*. We give the matrix K a small perturbation U_i and roll out the policy to collect costs and states. Then the estimated gradients can be described by:

$$\widehat{\nabla C(K)} = \frac{1}{m}\sum_{i=1}^{m}\frac{d}{r^2}\widehat{C}_i U_i,$$

where m is the number of samples; d is the dimension and r is the norm of the perturbation. The estimates come from zero-order optimization:

$$\nabla f_{\sigma^2}(x) = \frac{1}{\sigma^2}\mathbb{E}_{\varepsilon\sim\mathcal{N}(0,\sigma^2\mathrm{I})}[f(x+\varepsilon)\varepsilon]$$

Σ_K can be also estimated by sampling.

The paper proves that this model-free, policy gradient methods also lead to globally optimal policies, with both polynomial computational and sample complexities.

4 PROOF SKETCH

In this part I will briefly summarize the proof idea according to my own understandings. Polyak (1963) points out that gradient domination plus smoothness imply linear convergence to global optima even in non-convex settings. *Therefore, the two key ingredients of proof are gradient domination and smoothness analysis.*

The good news is that LQR satisfies the gradient domination condition, which can be described by the following theorem:

Theorem 2 *(Gradient domination) Let K^* be an optimal policy. Suppose K has finite cost and $\sigma_{\min}(\Sigma_K) > 0$. It holds that*

$$C(K) - C(K^*) \leq \frac{\|\Sigma_{K^*}\|}{\sigma_{\min}(\Sigma_K)^2\,\sigma_{\min}(R)}\|\nabla C(K)\|_F^2$$

The bad news is that the LQR objective cannot satisfy the smoothness condition globally which is because the objective becomes infinity when the matrix $A-BK$ becomes unstable. At the boundary between stable and unstable policies, the objective function quickly becomes infinity, which violates the traditional smoothness conditions. However, the paper proves that the objective of LQR satisfy an almost-smoothness condition. It is this almost-smooth property that leads to global convergence.

Theorem 3 *("Almost" smoothness) $C(K)$ satisfies*

$$\begin{aligned} C(K) - C(K') = &-2\,\mathrm{Tr}(\Sigma_{K'}(K-K')^T E_K) \\ &+ \mathrm{Tr}(\Sigma_{K'}(K-K')^T(R+B^T P_K B)(K-K')) \end{aligned}$$

The next step is to prove convergence of the model-free case. There are two key points.

- Since the algorithm only rolls out finite steps for this LQR problem of infinite horizons, the paper proves that when the roll out length l is large enough, the cost function C and the covariance $\sum$ are approximately equal to the corresponding quantities at infinite steps.
- Since gradients are estimated by sampling, the paper proves this estimation is within desired accuracy

5 IMPLEMENTATION

In this section we will report results of our implementation for the model-free algorithm. The source code can be found at https://github.com/xiangyu-liu/PG4LQR. The algorithm given by the original

paper just provides with method for estimating gradients. Therefore, I choose to adopt the common optimization algorithm, Adam (Kingma & Ba, 2014) to optimize the policy. We also adopt another trick to stablize the training, which is clipping the gradient when it is too large.

Figure 1 displays our simulation results. We can find that the gap between current policy and the optimal policy becomes small quickly. Within 5000 iterations, $\frac{C(K)-C(K^*)}{C(K^*)}$ can be less than 0.2. However, compared with the simulation results of exact gradient descent reported by the original paper there is still a huge gap in terms of costs and the convergence speed since our simulation is model-free. Another important finding is that although natural policy gradient methods enjoy better theoretical properties the experimental performance is worse than simple gradient descent. The gap becomes more larger when the dimension of the system grows. One possible reason is that the estimate of Σ_K is not accurate enough. However, if we increase the roll-out length, the efficiency will become much worse. Currently, I have not found appropriate hyper-parameters such that the natural policy gradient methods can converge to a decent level when the state dimension is 100 and action dimension is 20, which is why I do not plot the corresponding line in the second plot. Therefore, I am quite suspicious about the practical use of natural policy gradient methods. Besides, if it is also worthwhile to mention that one drawback of this algorithm is that it is quite sensitive to hyper-parameters. Without appropriate hyper-parameters, the algorithm cannot even converge.

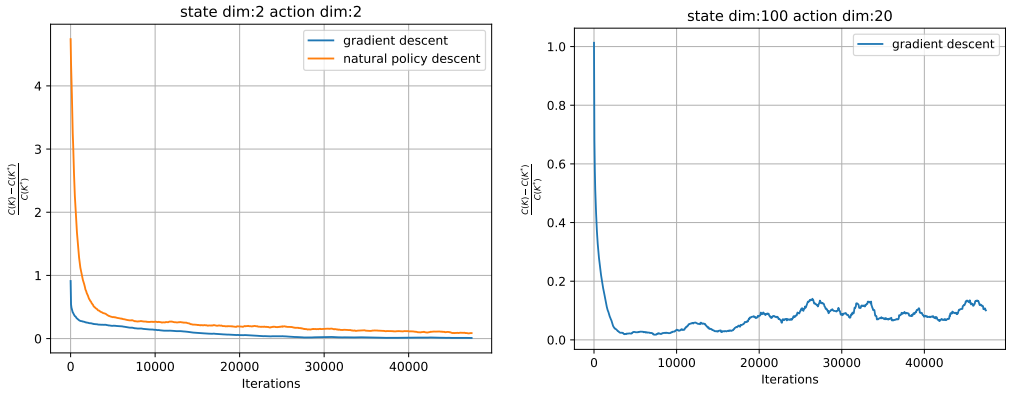

Figure 1: This figure shows the gap between the current policy and the optimal policy during training. The left one shows a system whose state dimension is 2 and action dimension is 2. The right one shows a system whose state dimension is 100 and action dimension is 20

6 Summary

This work has provided provable guarantees that model-based gradient methods and model-free policy gradient methods converge to the globally optimal solution, with finite polynomial computational and sample complexities, bridging works between optimal control theory and sample based reinforcement learning methods. We have implemented the model-free policy gradient algorithm and verify its effectiveness.

References

Brian DO Anderson and John B Moore. *Optimal control: linear quadratic methods*. Courier Corporation, 2007.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Boris T Polyak. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4):864–878, 1963.