International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 3, Number 1, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v3n1.02
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Review of Two-timescale Stochastic Approximation Algorithms

Jingjing Liang, Qingtao Wu

School of Information Engineering, Henan University of Science and Technology, Luoyang 471023, China

ABSTRACT

Artificial Intelligence has gradually become an important force to drive human beings into the intelligent era, and machine learning has made great contributions to the rise and development of Artificial Intelligence. Stochastic approximation (SA) is a com-monly used optimization algorithm in machine learning, and with the complexity of practical problem scenarios, two-timescale SA have received extensive attention and research. In this paper, the basic idea and development process of SA are introduced firstly, followed by the description of several algorithmic frameworks for linear and nonlinear SA, and the specific applications of two-timescale SA in the fields of opti-mization and reinforcement learning are also introduced. Finally, the two-timescale SA is summarized and outlooked.

KEYWORDS

Stochastic Approximation; Two-timescale; Optimization Algorithm

1. INTRODUCTION

In the field of artificial intelligence, machine learning has become the method of choice for developing software for speech recognition, computer vision, robot control, natural language processing [1]. Optimization algorithms [2] are one of the core components of machine learning, and the essence of most machine learning algorithms is to build an optimization model and learn the parameters of the objective function from observed data samples. In the era of massive data, the performance of optimization algorithms greatly affects the promotion and application of machine learning models. Stochastic approximation (SA), originally proposed by Robbins and Monro[3], is one of the most popular methods for solving a variety of optimization problems, where values of objective functions are only observed randomly.

The field of stochastic approximation is growing rapidly, affecting application areas from signal processing to artificial intelligence [4]. The problem of maximizing a function using SA was first presented in [5]. The method of using ordinary differential equations to follow the trajectory of the SA process was presented in [6, 7] and its convergence to the stability limit was proved. However, in many coupled applications, single-timescale SA is often time-consuming or even impossible to complete model training [8, 9]. In this case, it is crucial to design two-timescale SA algorithms with respect to the coupled updates of the two parameters.

Therefore, the continuous research on two-timescale SA algorithms has important research implications for machine learning optimization algorithms and even for the field of artificial intelligence.

2. CLASSICAL STOCHASTIC APPROXIMATION METHODS

2.1. Linear Singer-timescale Stochastic Approximation

The goal of linear singer-timescale SA is to find a solution θ^* for a linear system. To obtain θ^*, one can iterate through the following recursion:

$$\theta_{t+1} = \theta_t - \alpha_t(A_{t+1}\theta_t - b_{t+1} + \zeta_t) \tag{1}$$

where $\{(A_t, b_t) : t \in \mathbb{N}^*\}$ is a sequence of random variables and variables $\bar{A} \in \mathbb{R}^{d\times d}$ and $\bar{b} \in \mathbb{R}^d$ are usually unknown. Here, ζ_t is the sampling noise sequence and α_t is the step size for the iterative update of the algorithm.

The recursive formula (1) is referred to as linear singer-timescale SA in [10,11]. This class of algorithms and their associated settings have a long history and a wide range of applications in the field of signal processing [12]. It reignited interest in computational statistics and machine learning especially in optimization algorithms, reinforcement learning (RL) and Q-learning [13-15]. Linear singer-timescale SA has been intensively studied in several works to establish theoretical guarantees for its asymptotic [16] and non-asymptotic [17] convergence analysis.

2.2. Nonlinear Singer-timescale Stochastic Approximation

The goal of the nonlinear singer-timescale SA is to find a solution or extreme value for the nonlinear system $F(\cdot)$ when $F(\cdot)$ is unknown and the value of the $F(\cdot)$ cannot be measured without error. In other words, the algorithm needs to find a point θ^* of the unknown nonlinear function $F(\theta)$ such that $F(\theta^*) = 0$. Thus, the point θ^* can be approximated by iteratively updating the variable θ_t as follows:

$$\theta_{t+1} = \theta_t - \alpha_t F(\theta_t, \zeta_t) \tag{2}$$

Furthermore, the convergence analysis of the nonlinear singer-timescale SA algorithm under the Markovian noise model was carried out in [18]. The parameter updating via stochastic gradient descent and its variations is discussed in references [19, 20].

2.3. Linear Two-timescale Stochastic Approximation

Consider the problem of finding the solution (θ^*, ω^*) of the following linear system of equations:

$$\begin{aligned} A_{11}\theta^* + A_{12}\omega* = b1, \\ A_{21}\theta^* + A_{22}\omega* = b2 \end{aligned} \tag{3}$$

Where the set of matrices $A_{i,j}$,$i, j = 1, 2$ is not available and can only be sampled from some uncertain data. Therefore, it is infeasible to compute (3) directly. In addition, the researchers have considered iterative methods for solving this problem. In particular, there is interest in using a linear two time-scale SA with iteratively updated estimates (θ^t, ω^t) of (θ^*, ω^*)as follows:

$$\begin{aligned} \theta_{t+1} = \theta_t - \alpha_t \left(A_{11}\theta^t + A_{12}\omega_t - b1 + \zeta_t\right), \\ \omega_{t+1} = \omega_t - \gamma_t \left(A_{12}\theta^t + A_{22}\omega_t - b2 + \psi_t\right) \end{aligned} \tag{4}$$

Here, α_t and γ_t are two nonnegative step sizes, ζ_t and ψ_t are sequences of sampling noise. Most of the research on the asymptotic and non-asymptotic convergence of linear two-timescale SA algorithms has been in optimization [21, 22] and reinforcement learning [23, 24].

2.4. Nonlinear Two-timescale Stochastic Approximation

The goal of nonlinear two-timescale SA is to find the solution of two coupled unknown nonlinear function systems $F(\cdot,\cdot)$ and $G(\cdot,\cdot)$. For two nonlinear functions $F:\mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}^d$ and $G:\mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}^d$, it find $\theta^* \text{ and } \omega^*$ such that:

$$\begin{cases} F(\theta^*, \omega^*) = 0, \\ G(\theta^*, \omega^*) = 0. \end{cases} \tag{5}$$

In fact, because $F(\cdot,\cdot)$ and $G(\cdot,\cdot)$ are unknown, the method can obtain corresponding values $F(\theta,\omega)+\zeta$ and $G(\theta,\omega)+\psi$ for each of (θ,ω), where ζ and ψ are random variables generated by a stochastic oracle. The parameter update step of the two-timescale SA is:

$$\begin{aligned} \theta_{t+1} &= \theta_t - \alpha_t \left(F(\theta_t, \omega_t) + \zeta_t\right), \\ \omega_{t+1} &= \omega_t - \gamma_t \left(G(\theta_t, \omega_t) + \psi_t\right) \end{aligned} \tag{6}$$

Where α_t and γ_t are two nonnegative step sizes such that $\gamma_t \ll \alpha_t$. The asymptotic and non-asymptotic convergence analysis method for nonlinear two-timescale SA algorithms is studied in [25-27].

3. MOTIVATING APPLICATIONS

3.1. SGD with Polyak-Ruppert Averaging

It is assumed that a function f is to be minimized, but only a noisy estimate of the true gradient of the parameter can be obtained. To find the true minimizer, the stochastic gradient method [28] iteratively updates the parameter ω_t. To improve the convergence of stochastic gradient methods, an additional averaging step [29, 30] is often used. The update steps of the algorithm are as follows:

$$\begin{aligned} \omega_{t+1} &= \omega_t - \alpha_t \left(\nabla f\left(\omega_t\right) + \psi_t\right), \\ \theta_{t+1} &= \frac{1}{t+1}\sum_{t=0}^{t} \omega_t = \theta_t + \frac{1}{t+1}\left(\omega_t - \theta_t\right) \end{aligned} \tag{7}$$

Obviously, two updates are a special case of the two-timescale SA in (7) with $F(\theta,\omega) = \nabla f(\omega)$ and $G(\theta,\omega) = \omega - \theta$.

3.2. SGD with Momentum

Stochastic heavy ball [31] is a variant of the stochastic gradient method based on momentum and adaptive step sizes and has been shown to be effective. The iterative relationship for the normalized version of stochastic heavy ball [32, 33] is:

$$\omega_{t+1} = \omega_t - \alpha_t\left(\omega_t - \nabla f\left(\theta_t\right) - \psi_t\right),$$
$$\theta_{t+1} = \theta_t - \beta_t \omega_t \quad (8)$$

Here one should interpret ω_t as a (stochastic) search direction that is defined to be a combination of the current stochastic gradient $\nabla f\left(\theta_t\right) + \psi_t$ and past search direction ω_t. These two updates are a special case of the two-timescale SA in (7) with $F(\theta,\omega) = \omega - \nabla f(\theta)$ and $G(\theta,\omega) = \theta$.

3.3. Minimax Optimization

Consider the following minimax optimization problems[34], where f is a (non)convex function with respect to θ when for a fixed ω and (non) concave with respect to ω when for a fixed θ. This method iteratively updates the estimates θ_t and ω_t of the desired solution as:

$$\theta_{t+1} = \theta_t - \alpha_t \nabla_\theta f(\theta_t, \omega_t; \zeta_t),$$
$$\omega_{t+1} = \omega_t - \beta_t \nabla_\omega f(\theta_t, \omega_t; \psi_t)) \quad (9)$$

where α_t and β_t are chosen differently to ensure the convergence of the method. The minimax problem has broad applications in different areas including training generative adversarial networks [35], adversarial and robust machine learning [36], and distributed optimization [37].

3.4. GTD Learning

In reinforcement learning, GTD learning [39] can be used to solve the policy evaluation problem under the approximation of nonlinear functions, which can be regarded as a specific case of two-timescale SA. The goal of policy evaluation is to estimate the cumulative reward obtained with the policy π. The iteration of the GTD algorithm is described as follows:

$$\theta_{t+1} = \theta_t + \alpha_t\left(\left(\phi_t - \gamma\phi_t'\right)\left(\phi_t^\top \omega_t\right) - h_t\right),$$
$$\omega_{t+1} = \omega_t + \gamma_t\left(\delta_t - \phi_t^\top \omega_t\right)\phi_t \quad (10)$$

Where $\delta_t = r\left(s_t, a_t\right) + \gamma V_{\theta_t}\left(s_{t+1}\right) - V_{\theta_t}\left(s_t\right)$ and $h_t = \left(\delta_t - \phi_t^\top \omega_t\right)\nabla^2 V_{(\theta_t)}(s_t)\omega_t$.

4. SUMMARY

Although the two-timescale SA algorithm has been applied in many fieldss, its application is still broad. In future work, it can be further studied from the following two points: Most of the current research on two-timescale SA algorithms is based on the ideal assumption that the objective function is smooth, which is not the case in reality. If the objective function established about the model is non-smooth, the problem will become more complicated, which will be very worthy of consideration in future research. Secondly, it is also worth exploring how to extend the two-timescale SA algorithm to distributed architectures with a wider range of applications.

ACKNOWLEDGMENTS

This work was supported in part by the Key Technologies R & D Program of Henan Province under Grant No.222102210080, 232102211008 and 242102210102.

REFERENCES

[1] Jordan M I, Mitchell T M. Machine learning: Trends, perspectives, and prospects [J]. Science, 2015, 349(6245): 255-260.

[2] Sun S, Cao Z, Zhu H, et al. A survey of optimization methods from a machine learning perspective [J]. IEEE Transactions on Cybernetics, 2019, 50(8): 3668-3681.

[3] Robbins H, Monro S. A stochastic approximation method [J]. The Annals of Mathematical Statistics, 1951: 400-407.

[4] Vajjha K, Trager B, Shinnar A, et al. Formalization of a stochastic approximation theorem [C]. Proceedings of the 13th International Conference on Interactive Theorem Proving. 2022, 31:1-18.

[5] Kiefer J, Wolfowitz J. Stochastic estimation of the maximum of a regression function [J]. The Annals of Mathematical Statistics, 1952: 462-466.

[6] Ljung L. Analysis of recursive stochastic algorithms [J]. IEEE Transactions on Automatic Control, 1977, 22(4): 551-575.

[7] Kushner H J, Clark D S. Stochastic approximation methods for constrained and unconstrained systems [M]. Springer Science and Business Media, 2012.

[8] Wang M, Fang E X, Liu H. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions [J]. Mathematical Programming, 2017, 161: 419-449.

[9] Hu W, Li C J, Lian X, et al. Efficient smooth non-convex stochastic compositional optimization via stochastic recursive gradient descent[J]. Advances in Neural Information Processing Systems, 2019, 32.

[10] Lakshminarayanan C, Szepesvari C. Linear stochastic approximation: How far does constant step-size and iterate averaging go? [C]. International Conference on Artificial Intelligence and Statistics. PMLR, 2018: 1347-1355.

[11] Durmus A, Moulines E, Naumov A, et al. Tight high probability bounds for linear stochastic approximation with fixed stepsize [C]. Proceedings of the 35th Conference on Neural Information Processing Systems, 2021, 34: 30063-30074.

[12] Brooms A C. Stochastic approximation and recursive algorithms with applications, 2nd edn by hj kushner and gg yin [J]. 2006.

[13] Bottou L, Curtis F E, Nocedal J. Optimization methods for large-scale machine learning [J]. SIAM Review, 2018, 60(2): 223-311.

[14] Mou W, Pananjady A, Wainwright M J, et al. Optimal and instance-dependent guarantees for Markovian linear stochastic approximation [C]. Proceedings of Machine Learning Research, PMLR, 2022, 2060–2061.

[15] Clifton J, Laber E. Q-learning: Theory and applications [J]. Annual Review of Statistics and Its Application, 2020, 7: 279-301.

[16] Borkar V S. Stochastic approximation: A dynamical systems viewpoint [M]. Springer, 2009.

[17] Srikant R, Ying L. Finite-time error bounds for linear stochastic approximation and td learning [C]. Proceedings of the 32nd Conference on Learning Theory. PMLR, 2019: 2803-2830.

[18] Qu G, Wierman A. Finite-time analysis of asynchronous stochastic approximation and Q-learning [C]. Proceedings of the 33rd Conference on Learning Theory. PMLR, 2020: 3185-3205.

[19] Sun T, Sun Y, Yin W. On markov chain gradient descent [C]. Proceedings of the 32nd Conference on Neural Information Processing Systems, 2018, 31.

[20] Zhang L, Zhou Z H. Stochastic approximation of smooth and strongly convex functions: Beyond the $O(1/t)$ convergence rate [C]. Proceedings of the 32nd Conference on Learning Theory. PMLR, 2019: 3160-3179.

[21] Doan T T, Romberg J. Linear two-time-scale stochastic approximation a finite-time analysis [C]. Proceedings of the 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton). Monticello, IL, USA, 2019: 399-406.

[22] Doan T T. Finite-time analysis and restarting scheme for linear two-time-scale stochastic approximation [J]. SIAM Journal on Control and Optimization, 2021, 59(4): 2798-2819.

[23] Dalal G, Thoppe G, Szörényi B, et al. Finite sample analysis of two-timescale stochastic approximation with applications to reinforcement learning [C]. Proceedings of the 31st Conference On Learning Theory. PMLR, 2018: 1199-1233.

[24] Kaledin M, Moulines E, Naumov A, et al. Finite time analysis of linear two-timescale stochastic approximation with Markovian noise [C]. Proceedings of 33rd Conference on Learning Theory. PMLR, 2020: 2144-2203.

[25] Mokkadem A, Pelletier M. Convergence rate and averaging of nonlinear two-time-scale stochastic approximation algorithms [J]. The Annals of Applied Probability, 2006: 1671-1701.

[26] Chung W, Nath S, Joseph A, et al. Two-timescale networks for nonlinear value function approximation [C]. Proceedings of the International Conference on Learning Representations. 2019.

[27] Doan T T. Nonlinear two-time-scale stochastic approximation convergence and finite-time performance [J]. IEEE Transactions on Automatic Control, 2022.

[28] Amir I, Koren T, Livni R. SGD generalizes better than GD (and regularization doesn't help) [C]. Proceedings of 34th Conference on Learning Theory. USA: PMLR, 2021: 63-92.

[29] Polyak B T, Juditsky A B. Acceleration of stochastic approximation by averaging [J]. SIAM Journal on Control and Optimization, 1992, 30(4): 838-855.

[30] Mou W, Li C J, Wainwright M J, et al. On linear stochastic approximation: Fine-grained Polyak-Ruppert and non-asymptotic concentration [C]. Proceedings of 33rd Conference on Learning Theory. PMLR, 2020: 2947-2997.

[31] Gadat S, Panloup F, Saadane S. Stochastic heavy ball [J]. Electronic Journal of Statistics, 12:461–529, 2018.

[32] Gitman I, Lang H, Zhang P, et al. Understanding the role of momentum in stochastic gradient methods [C]. Advances in Neural Information Processing Systems, 2019, 32.

[33] Han Y, Li X, Zhang Z. Finite-time decoupled convergence in nonlinear two-time-scale stochastic approximation [J]. arXiv preprint arXiv:2401.03893, 2024.

[34] Huang F, Wu X, Hu Z. Adagda: Faster adaptive gradient descent ascent methods for minimax optimization [C]. International Conference on Artificial Intelligence and Statistics. PMLR, 2023: 2365-2389.

[35] Creswell A, White T, Dumoulin V, et al. Generative adversarial networks: An overview [J]. IEEE signal processing magazine, 2018, 35(1): 53-65.

[36] Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale [J]. arXiv preprint arXiv:1611.01236, 2016.

[37] Lan G, Lee S, Zhou Y. Communication-efficient algorithms for decentralized and stochastic optimization [J]. Mathematical Programming, 2020, 180(1): 237-284.

[38] Maei H, Szepesvari C, Bhatnagar S, et al. Convergent temporal-difference learning with arbitrary smooth function approximation [C]. Proceedings of the 22nd International Conference on Neural Information Processing Systems, 2009: 1204–1212.