

Stochastic-Constrained Stochastic Optimization with Markovian Data

Yeongjong Kim

Center for Mathematical Machine Learning and its Applications (CM2LA)
Postech

NeurIPS 2025

Joint work with Dabeen Lee (SNU)

- ▶ Stochastic optimization with stochastic constraint :

$$\min_{x \in \mathcal{X}} \bar{f}(x) := \mathbb{E}_{\xi \sim \mu}[f(x, \xi)] \quad \text{subject to} \quad \bar{g}(x) := \mathbb{E}_{\xi \sim \mu}[g(x, \xi)] \leq 0,$$

where f, g are convex.

- ▶ However, we cannot access μ directly, only an ergodic Markov chain $\{\xi_t\}$ whose stationary distribution is μ .

- 1 Ergodic Markov chain
- 2 Stochastic Optimization with Markovian data
- 3 Stochastic constrained stochastic optimization with Markovian data

- 1 Ergodic Markov chain
- 2 Stochastic Optimization with Markovian data
- 3 Stochastic constrained stochastic optimization with Markovian data

Definition

A Markov chain with transition matrix P is called **ergodic** if

- (1) it has a unique stationary distribution μ i.e. $\mu P = \mu$,
- (2) $\lim_{t \rightarrow \infty} \sup_{\nu} \|\nu P^t - \mu\|_{TV} = 0$.

► We define

$$d_{\text{mix}}(t) := \sup_{\nu} \|\nu P^t - \mu\|_{TV}$$

and the **mixing time**

$$\tau_{\text{mix}}(\epsilon) := \inf\{t \in \mathbb{N} : d_{\text{mix}}(t) \leq \epsilon\},$$

$$\tau_{\text{mix}} := \tau_{\text{mix}}(1/4).$$

► It is known that

$$\tau_{\text{mix}}(\epsilon) \leq \lceil \log_2 \frac{1}{\epsilon} \rceil \tau_{\text{mix}}.$$

- ▶ i.i.d. sequence is a special case of ergodic Markov chain with $\tau_{\text{mix}}(\epsilon) = 1$.
- ▶ Let $\{\xi_t\}$ be an ergodic Markov chain and $\epsilon > 0$ be small enough. Then the samples with $\tau_{\text{mix}}(\epsilon)$ time differences are almost i.i.d.

$$\|\nu P^{\tau_{\text{mix}}(\epsilon)} - \mu\| \leq \epsilon$$

$$\xi_t \xrightarrow{\tau_{\text{mix}}(\epsilon)} \xi_{t+\tau_{\text{mix}}(\epsilon)} \xrightarrow{\tau_{\text{mix}}(\epsilon)} \xi_{t+2\tau_{\text{mix}}(\epsilon)} \rightarrow \dots$$

Why Markov chain?

- ▶ We want to consider dependent data sequence.
- ▶ In many cases, we can not directly access to the target distribution μ . Instead, we can construct a Markov chain converging to μ (**Markov Chain Monte Carlo** method).

- 1 Ergodic Markov chain
- 2 Stochastic Optimization with Markovian data
- 3 Stochastic constrained stochastic optimization with Markovian data

Stochastic Optimization (without constraint) with Markovian data

- ▶ Stochastic optimization :

$$\min_{x \in \mathcal{X}} \bar{f}(x) := \mathbb{E}_{\xi \sim \mu}[f(x, \xi)],$$

where f is convex.

- ▶ However, we cannot access μ directly, only an ergodic Markov chain $\{\xi_t\}$ whose stationary distribution is μ .

Stochastic Optimization with Markovian data

- ▶ At t , we choose $x_t \in \mathcal{X}$, without knowing $f_t := f(x, \xi_t)$.
- ▶ A convex function f_t is sampled from ξ_t and we get some information about f_t at x_t : $f_t(x_t)$ and $\nabla f_t(x_t)$

$$x_1 \rightarrow f_1 \rightarrow \cdots \rightarrow x_t \rightarrow f_t \rightarrow \cdots \rightarrow x_T \rightarrow f_T$$

- ▶ Goal : We want to minimize the term

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)],$$

where $\bar{x}_T = \frac{1}{T} \sum_{t=1}^T x_t$, $x^* = \operatorname{argmin}_{x \in \mathcal{X}} \bar{f}(x)$.

Ergodic Mirror Descent (Duchi et al., 2012)

- ▶ (GD form) We update $x_{t+1} = \mathcal{P}_{\mathcal{X}}(x_t - \alpha_t \nabla f_t(x_t))$.
- ▶ Under F -Lipschitz condition of f , bounded domain \mathcal{X} (of diameter R),

Theorem (Duchi et al., 2012)

EMD with $\alpha_t = \frac{R}{F\sqrt{\tau_{\text{mix}}(T^{-1/2})t}}$ gives

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] = \mathcal{O}\left(\sqrt{\frac{\tau_{\text{mix}}(T^{-1/2})}{T}}\right) = \tilde{\mathcal{O}}\left(\sqrt{\frac{\tau_{\text{mix}}}{T}}\right).$$

- ▶ We need to know the mixing time $\tau_{\text{mix}}(T^{-1/2})$ in order to choose α_t .

- ▶ (MLMC sampling) Now we get \mathbf{N}_t samples at each time t .

$$x_t \rightarrow f_t^{(1)}, \dots, f_t^{(N_t)} \rightarrow x_{t+1} \rightarrow \dots$$

Here, N_t itself is a random variable $N_t = \begin{cases} 2^{J_t}, & \text{if } 2^{J_t} \leq T \\ 1, & \text{if } 2^{J_t} > T \end{cases}$

where $J_t \sim \text{Geom}(1/2)$.

- ▶ After getting $f_t^{(1)}, \dots, f_t^{(N_t)}$, we define **the estimator \mathbf{f}_t** of \bar{f} as
$$f_t = f_t^1 + \begin{cases} 0, & \text{if } N_t = 1 \\ N_t(f_t^{N_t} - f_t^{N_t/2}), & \text{if } N_t \geq 2, \end{cases} \text{ where } f_t^N = \frac{1}{N} \sum_{i=1}^N f_t^{(i)}(x_t).$$

- ▶ (AdaGrad) Then we update using
$$x_{t+1} = \mathcal{P}_{\mathcal{X}}(x_t - \eta_t \nabla f_t(x_t)); \quad \eta_t = \frac{\alpha}{\sqrt{\sum_{k=1}^t \|\nabla f_k(x_k)\|^2}}.$$

- ▶ Under Lipschitzness & boundedness condition of f , bounded domain \mathcal{X} (of diameter R),

Theorem (Dorfman et al., 2022)

MAG with $\alpha = R/\sqrt{2}$ gives

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] = \tilde{\mathcal{O}}\left(\sqrt{\frac{\tau_{mix}}{T}}\right).$$

- ▶ MAG does not require the knowledge of mixing time.
- ▶ Instead, each iteration requires $\mathbb{E}[N_t] = \mathcal{O}(\log T)$ samples in expectation.

- 1 Ergodic Markov chain
- 2 Stochastic Optimization with Markovian data
- 3 Stochastic constrained stochastic optimization with Markovian data

Q. Can we get similar results for constrained setting?

A. The answer is yes, although the extension to the constrained case is not trivial.

- ▶ Stochastic optimization with stochastic constraint :

$$\min_{x \in \mathcal{X}} \bar{f}(x) := \mathbb{E}_{\xi \sim \mu}[f(x, \xi)] \quad \text{subject to} \quad \bar{g}(x) := \mathbb{E}_{\xi \sim \mu}[g(x, \xi)] \leq 0,$$

where f, g are convex.

- ▶ We cannot access μ directly, but only an ergodic Markov chain $\{\xi_t\}$ whose stationary distribution is μ .

- ▶ At every time step t , we choose a point $x \in \mathcal{X}$, without knowing $f_t := f(x, \xi_t), g_t := g(x, \xi_t)$.
- ▶ Convex functions f_t, g_t are sampled from ξ_t and we get some information about f_t, g_t at x_t : $f_t(x_t), \nabla f_t(x_t), g_t(x_t), \nabla g_t(x_t)$

$$x_1 \rightarrow f_1, g_1 \rightarrow \cdots \rightarrow x_t \rightarrow f_t, g_t \rightarrow \cdots \rightarrow x_T \rightarrow f_T, g_T$$

- ▶ Goal : We want to bound both

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] \quad \text{and} \quad \mathbb{E}[\bar{g}(\bar{x}_T)],$$

where $\bar{x}_T = \frac{1}{T} \sum_{t=1}^T x_t$, $x^* = \operatorname{argmin}_{x \in \mathcal{X}, \bar{g}(x) \leq 0} \bar{f}(x)$.

Drift-Plus-Penalty Algorithm (Yu et al., 2017)

- ▶ Pick $x_1 \in \mathcal{X}$, $Q_1 = 0$.
- ▶ Update rule :

$$x_{t+1} = \operatorname{argmin}_{x \in \mathcal{X}} \left\{ (V_t \nabla f_t(x_t) + Q_t \nabla g_t(x_t))^\top x + \alpha \|x - x_t\|_2^2 \right\}$$
$$Q_{t+1} = \max\{Q_t + g_t(x_t) + \nabla g_t(x_t)^\top (x_{t+1} - x_t), 0\}.$$

Drift-Plus-Penalty Algorithm (Yu et al., 2017)

- ▶ Pick $x_1 \in \mathcal{X}$, $Q_1 = 0$.
- ▶ Update rule :

$$x_{t+1} = \operatorname{argmin}_{x \in \mathcal{X}} \left\{ (V_t \nabla f_t(x_t) + Q_t \nabla g_t(x_t))^\top x + \alpha \|x - x_t\|_2^2 \right\}$$
$$Q_{t+1} = \max \left\{ Q_t + \underbrace{g_t(x_t) + \nabla g_t(x_t)^\top (x_{t+1} - x_t)}_*, 0 \right\}.$$

- ▶ If we remove *, it is just the standard first-order primal-dual algorithm

$$x_{t+1} = P_{\mathcal{X}} \left[x_t - \frac{1}{2\alpha} \nabla_x L_t(x_t, Q_t) \right]$$
$$Q_{t+1} = P_{\mathbb{R}_{\geq 0}} [Q_t + \nabla_Q L_t(x_t, Q_t)],$$

where $L_t(x, Q) = V_t f_t(x) + Q g_t(x)$.

Algorithm 1 - EDPP

By taking mixing-time-aware choices for V_t and α_t , we get our first algorithm below.

Algorithm 1 Ergodic Drift-Plus-Penalty (EDPP)

Initialize: Initial iterates $\mathbf{x}_1 \in \mathcal{X}$, $Q_1 = 0$, and $0 < \beta \leq 1/2$.

for $t = 1$ **to** T **do**

Observe f_t and g_t .

Set penalty parameter V_t and step size parameter α_t as

$$V_t = (\tau_{\text{mix}} t)^\beta, \quad \alpha_t = \tau_{\text{mix}} t.$$

Primal update: Set \mathbf{x}_{t+1} as

$$\mathbf{x}_{t+1} = \operatorname{argmin}_{\mathbf{x} \in \mathcal{X}} \left\{ (V_t \nabla f_t(\mathbf{x}_t) + Q_t \nabla g_t(\mathbf{x}_t))^\top \mathbf{x} + \alpha_t D(\mathbf{x}, \mathbf{x}_t) \right\}$$

Dual update: Set Q_{t+1} as

$$Q_{t+1} = \left[Q_t + g_t(\mathbf{x}_t) + \nabla g_t(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t) \right]_+$$

end for

Algorithm 1 - EDPP

Under standard assumptions (Lipschitzness & boundedness of f, g , boundedness of \mathcal{X}), we have that

Theorem 1

EDPP gives

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] = \tilde{O}\left(\frac{\tau_{mix}^{1-\beta}}{T^\beta} + \frac{\tau_{mix}^{\beta/2}}{T^{(1-\beta)/2}}\right)$$
$$\mathbb{E}[\bar{g}(\bar{x}_T)] = \tilde{O}\left(\frac{\tau_{mix}^{\beta/2}}{T^{(1-\beta)/2}}\right).$$

Algorithm 2 - DPP-DD

Algorithm 2 below samples data every mixing-time steps to reduce correlation.

Algorithm 2 Drift-Plus-Penalty with Data Drop (DPP-DD)

Initialize: Initial iterates $\mathbf{x}_1 \in \mathcal{X}$, $Q_1 = 0$, and $0 < \beta \leq 1/2$.

for $k = 1$ **to** $\lfloor T/\tau_{\text{mix}} \rfloor$ **do**

Set $\mathbf{x}_{(k-1)\tau_{\text{mix}}+i} = \mathbf{x}_{(k-1)\tau_{\text{mix}}+1}$ for $i = 2, \dots, \tau_{\text{mix}}$.

Observe $f_{k\tau_{\text{mix}}}$ and $g_{k\tau_{\text{mix}}}$.

Set penalty parameter V_k and step size parameter α_k as

$$V_k = k^\beta, \quad \alpha_k = k.$$

Primal update: Set $\mathbf{x}_{k\tau_{\text{mix}}+1}$ as

$$\mathbf{x}_{k\tau_{\text{mix}}+1} = \operatorname{argmin}_{\mathbf{x} \in \mathcal{X}} \left\{ (V_k \nabla f_{k\tau_{\text{mix}}}(\mathbf{x}_{k\tau_{\text{mix}}}) + Q_k \nabla g_{k\tau_{\text{mix}}}(\mathbf{x}_{k\tau_{\text{mix}}}))^\top \mathbf{x} + \alpha_k D(\mathbf{x}, \mathbf{x}_{k\tau_{\text{mix}}}) \right\}$$

Dual update: Set Q_{k+1} as

$$Q_{k+1} = \left[Q_k + g_{k\tau_{\text{mix}}}(\mathbf{x}_{k\tau_{\text{mix}}}) + \nabla g_{k\tau_{\text{mix}}}(\mathbf{x}_{k\tau_{\text{mix}}})^\top (\mathbf{x}_{k\tau_{\text{mix}}+1} - \mathbf{x}_{k\tau_{\text{mix}}}) \right]_+$$

end for

Algorithm 2 - DPP-DD

Under standard assumptions (Lipschitzness & boundedness of f, g , boundedness of \mathcal{X}), we have that

Theorem 2

DPP-DD gives

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] = \tilde{\mathcal{O}} \left(\frac{\tau_{mix}^\beta}{T^\beta} + \frac{\tau_{mix}^{(1-\beta)/2}}{T^{(1-\beta)/2}} \right)$$
$$\mathbb{E}[\bar{g}(\bar{x}_T)] = \tilde{\mathcal{O}} \left(\frac{\tau_{mix}^{(1-\beta)/2}}{T^{(1-\beta)/2}} \right).$$

- ▶ The convergence rates of EDPP and DPP-DD only differ in terms of τ_{mix} .

With Slater's condition

Definition

The constraint function \bar{g} is said to satisfy **Slater's condition** if there exist $\epsilon > 0, \hat{x} \in \mathcal{X}$ such that $\bar{g}(\hat{x}) \leq -\epsilon$.

Theorem 3

Under standard assumptions and Slater's condition, Both EDPP and DPP-DD with $\beta = 1/2$ give

$$\begin{aligned}\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] &= \tilde{O}\left(\sqrt{\frac{\tau_{\text{mix}}}{T}}\right) \\ \mathbb{E}[\bar{g}(\bar{x}_T)] &= \tilde{O}\left(\sqrt{\frac{\tau_{\text{mix}}}{T}}\right).\end{aligned}$$

- ▶ The previous two theorems require the knowledge of mixing time $\tau_{\text{mix}}(T^{-1})$ in the selection of V_t, α_t .

Algorithm 3 - MDP

- ▶ (MLMC sampling) We get \mathbf{N}_t samples at each time t .

$$x_t \rightarrow \begin{cases} f_t^{(1)}, \dots, f_t^{(N_t)} \\ g_t^{(1)}, \dots, g_t^{(N_t)} \end{cases} \rightarrow x_{t+1} \rightarrow \dots$$

Here, N_t itself is a random variable $N_t = \begin{cases} 2^{J_t}, & \text{if } 2^{J_t} \leq \mathbf{T}^2 \\ 1, & \text{if } 2^{J_t} > \mathbf{T}^2 \end{cases}$

where $J_t \sim \text{Geom}(1/2)$.

- ▶ After sampling, we define **the estimator** $\mathbf{f}_t, \mathbf{g}_t$ of \bar{f}, \bar{g} as before.
- ▶ (Adaptive DPP) We proposed AdaGrad-style variant of DPP

$$V_t = \Theta(S_{t-1}^\beta), \alpha_t = \Theta(S_{t-1})$$

where $S_t = \sum_{k=1}^t (\|\nabla f_t(x_t)\|^2 + \|\nabla g_t(x_t)\|^2 + |g_t(x_t)|^2)$.

Algorithm 3 - MDP

Algorithm 3 MLMC Adaptive Drift-Plus-Penalty (MDPP)

Initialize: Initial iterates $\mathbf{x}_1 \in \mathcal{X}$, $Q_1 = 0$ and parameters $0 < \beta \leq 1/2$, $\delta > 0$.

for $t = 1$ **to** T **do**

Observe f_t and g_t via MLMC method.

Set penalty parameter V_t , step size parameter α_t as (1).

Primal update: Set \mathbf{x}_{t+1} as

$$\mathbf{x}_{t+1} = \operatorname{argmin}_{\mathbf{x} \in \mathcal{X}} \left\{ (V_t \nabla f_t(\mathbf{x}_t) + Q_t \nabla g_t(\mathbf{x}_t))^\top \mathbf{x} + \alpha_t D(\mathbf{x}, \mathbf{x}_t) \right\}$$

Dual update: Set Q_{t+1} as

$$Q_{t+1} = \left[Q_t + g_t(\mathbf{x}_t) + \nabla g_t(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t) \right]_+$$

end for

What MLMC does

- ▶ MLMC estimator f_t satisfies

$$\mathbb{E}_{t-1}[f_t] = \mathbb{E}_{t-1} \left[\frac{f_t^{(1)} + \dots + f_t^{(T^2)}}{T^2} \right],$$

but the number of samples satisfies

$$\mathbb{E}[N_t] = \mathcal{O}(\log T).$$

- ▶ Its second moment

$$\mathbb{E}[\|\nabla f_t(x_t)\|^2], \quad \mathbb{E}[\|\nabla g_t(x_t)\|^2], \quad \mathbb{E}[g_t(x_t)^2] = \mathcal{O}(\tau_{\text{mix}})$$

combined with our adaptive method allows us to choose parameters as if we knew τ_{mix} without knowing it.

Algorithm 3 - MDPP

Under standard assumptions (Lipschitzness & boundedness of f, g , boundedness of \mathcal{X}), we have that

Theorem 4

MDPP gives

$$\mathbb{E}[\bar{f}(\bar{x}_T) - \bar{f}(x^*)] = \tilde{\mathcal{O}}\left(\frac{\tau_{mix}^{1-\beta}}{T^\beta}\right)$$
$$\mathbb{E}[\bar{g}(\bar{x}_T)] = \tilde{\mathcal{O}}\left(\frac{\tau_{mix}^{(2\beta+1)/4}}{T^{(1-\beta)/2}}\right).$$

- ▶ MDPP does not require the knowledge of mixing time.
- ▶ Instead, each iteration requires $\mathbb{E}[N_t] = \mathcal{O}(\log T)$ samples in expectation.

We conducted experiments on linear classification with fairness constraints using Markovian data. The following algorithms were evaluated for comparison:

- PD : Primal-dual method by Mahdavi et al. (2012).
- PD2 : Primal-dual method by Jenatton et al. (2016).
- DPP : Drift-plus-penalty algorithm by Yu et al. (2017).
- EDPP-t : Ergodic drift-plus-penalty (Algorithm 1).
- EDPP-T : modification of Algorithm 1 with non-adaptive parameters $V_t = \sqrt{\tau_{\text{mix}}T}$ and $\alpha_t = \tau_{\text{mix}}T$.
- MDPP : MLMC adaptive drift-plus-penalty (Algorithm 3).

Numerical Experiments

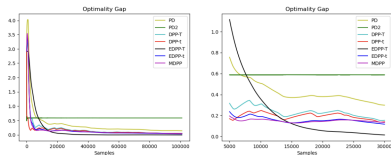


Figure 2: Optimalty Gap (Left), Enlarged Figure around 5,000 - 30,000 Samples (Right)

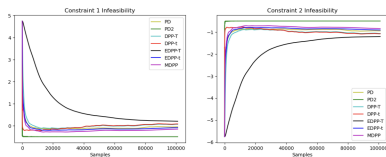





Figure 3: Constraint 1 Infeasibility (Left), Constraint 2 Infeasibility (Right)

Our algorithms outperform other methods.

Conclusion

- ▶ We are the first to show theoretical guarantees for stochastic constrained stochastic optimization with Markovian data.
- ▶ We showed that DPP achieves a good bound of the optimality gap and the infeasibility when we know the mixing time.
- ▶ If Slater's condition holds, DPP achieves better bounds.
- ▶ We proposed an AdaGrad-style variant of DPP, which is of independent interest.
- ▶ By combining our adaptive method with the MLMC estimator, we could achieve bounds comparable to those of DPP even though we do not know the mixing time.

-  J. Duchi et al., “Ergodic Mirror Descent,” *SIAM J. Optim.*, 2012.
-  R. Dorfman et al., “Adapting to Mixing Time in Stochastic Optimization with Markovian Data,” *ICML*, 2022.
-  H. Yu et al., “Online Convex Optimization with Stochastic Constraints,” *NeurIPS*, 2017.