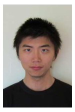
Analysis of Stochastic Optimization via Jump Systems and Quadratic Constraints



Anders Rantzer

joint with Bin Hu and Peter Seiler

LCCC Linnaeus Center
Lund University
Sweden

Outline

- **Integral Quadratic Constraints**
- Jump Dynamic Systems
- Numerical Rate Bounds for Stochastic Algorithms
- Analytical Rate Bounds
- Conclusions

Integral Quadratic Constraint

The (possibly nonlinear) operator Δ on $\ell_2^m[0,\infty)$ is said to *satisfy the IQC defined by* Π if

$$\int_0^{2\pi} \begin{bmatrix} \widehat{v}(e^{i\omega}) \\ \widehat{(\Delta v)}(e^{i\omega}) \end{bmatrix}^* \Pi(e^{i\omega}) \begin{bmatrix} \widehat{v}(e^{i\omega}) \\ \widehat{(\Delta v)}(e^{i\omega}) \end{bmatrix} d\omega \geq 0$$

for all $v \in \ell_2[0,\infty)$.

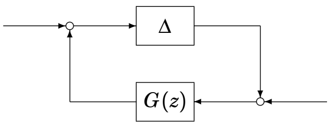
Example

If g is L-smooth and m-strongly convex, then

$$\begin{bmatrix} x - x^* \\ \nabla g(x) \end{bmatrix}^T \begin{bmatrix} -2mLI_p & (m+L)I_p \\ (m+L)I_p & -2I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \nabla g(x) \end{bmatrix} \geq 0$$

For non-convex g the inequality can be used with negative m.

IQC Stability Theorem



Let $G(s)$ be stable and proper and let Δ be causal.

For all $\tau \in [0,1]$, suppose the loop is well posed and $\tau\Delta$ satisfies the IQC defined by $\Pi(e^{i\omega})$. If

$$\begin{bmatrix} G(e^{i\omega}) \\ I \end{bmatrix}^* \Pi(e^{i\omega}) \begin{bmatrix} G(e^{i\omega}) \\ I \end{bmatrix} < 0 \qquad \text{for } \omega \in [0,\infty]$$

then the feedback system is input/output stable.

Used by Lessard, Recht, Packard (2015) to analyse deterministic algorithms.

Outline

- Integral Quadratic Constraints
- **Jump Dynamic Systems**
- Numerical Rate Bounds for Stochastic Algorithms
- Analytical Rate Bounds
- Conclusions

A Jump Linear System

Suppose that $i_1, i_2, \dots$ are identically, independently and uniformly distributed in the finite set $\mathcal{N} = \{1, \cdots, n\}$. Then, given matrix pairs $(A_1, B_1), \dots, (A_n, B_n)$, the dynamic system

$$\xi^{k+1} = A_{i_k}\xi^k + B_{i_k}w^k \qquad\qquad k \geq 1$$

is called a *jump linear system*.

Jump Linear System with Nonlinear Feedback

Suppose that the nonlinear map $\Delta : \xi \to w$ satisfies

$$\sum_{k=0}^{t} [C\xi^k + Dw^k]^T M [C\xi^k + Dw^k] \geq 0$$

for all solutions to $\xi^{k+1} = A_{i_k}\xi^k + B_{i_k}\Delta(\xi^k)$, $\xi^0 \in X_0$

Then

$$\mathbb{E}\|\xi^k\|^2 \leq \rho^{2k}\text{cond}(P)\|\xi^0\|^2$$

provided that $P \succ 0$ and

$$\frac{1}{n}\sum_{i=1}^{n} \begin{bmatrix} A_i^T P A_i - \rho^2 P & A_i^T P B_i \\ B_i^T P A_i & B_i^T P B_i \end{bmatrix} + \begin{bmatrix} C & D \end{bmatrix}^T M \begin{bmatrix} C & D \end{bmatrix} \prec 0$$

Outline

- Integral Quadratic Constraints
- Jump Dynamic Systems
- **Numerical Rate Bounds for Stochastic Algorithms**
- Analytical Rate Bounds
- Conclusions

Our Optimization Problem

Minimize

$$g(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

where $f_i : \mathbb{R}^p \to \mathbb{R}$ is L-smooth and g is m-strongly convex.

Empirical Risk Minimization

Many machine learning problems require optimizing an average loss over a finite training set:

$$\min_{x\in\mathbb{R}^p} g(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

- Ridge regression:
  $f_i(x) = (a_i^T x - b_i)^2 + \frac{m}{2}\|x\|^2$
- ℓ_2-regularized logistic regression:
  $f_i(x) = \log(1 + e^{-b_i a_i^T x}) + \frac{m}{2}\|x\|^2$
- ℓ_2-regularized loss minimization with loss function $l_i(x)$:
  $f_i(x) = l_i(x) + \frac{\lambda}{2}\|x\|^2$

Full Gradient Descent Method

- Gradient Descent Method
  $$x^{k+1} = x^k - \alpha\nabla g(x^k)$$
- Convergence is linear.
- Each iteration requires n computations:
  $\nabla g(x^k) = \frac{1}{n}\sum_{i=1}^{n}\nabla f_i(x^k)$

Stochastic Gradient Method

- [Robbins and Monro, 1951] used the iteration rule
  $$x^{k+1} = x^k - \alpha\nabla f_{i_k}(x^k)$$
  where the index i_k is randomly chosen for every k.
- Each iteration requires only one computation.
- With well-chosen constant step size, the method converges linearly to some tolerance of the optimum.

Stochastic Average Gradient (SAG) Method

[Roux et al., 2012; Schmidt et al., 2013] use the iteration rule:

$$x^{k+1} = x^k - \frac{\alpha}{n}\sum_{i=1}^{n} y_i^{k+1}$$

where at each iteration a random i_k is drawn and

$$y_i^{k+1} := \begin{cases} \nabla f_i(x^k) & \text{if } i = i_k \\ y_i^k & \text{otherwise} \end{cases}$$

Let $\alpha = \frac{1}{16L}$. Then $\mathbb{E}[g(x^k) - g(x^*)] \le C_0\left(1 - \min\{\frac{1}{8n}, \frac{m}{16L}\}\right)^k$.

Stochastic Finite-Sum Methods

- Now there is a large family of methods, e.g. SVRG, MISO, Finito, SDCA, and SAGA. Analysis is done case-by-case.
- For example, SAGA (Defazio et al., 2014) uses
  $$x^{k+1} = x^k - \alpha\left(\nabla f_{i_k}(x^k) - y_{i_k}^k + \frac{1}{n}\sum_{i=1}^{n} y_i^k\right)$$
  $$y_i^{k+1} = \begin{cases} \nabla f_i(x^k) & \text{if } i = i_k \\ y_i^k & \text{otherwise} \end{cases}$$
- SAGA and SAG look very similar. But the analysis of SAG is much more difficult! Why?

Finite-Sum Methods as Jump Systems

Finite-sum methods, e.g. SAGA, SAG, Finito, and SDCA, can be modeled jump dynamic systems:

$$\xi^{k+1} = A_{i_k}\xi^k + B_{i_k}w^k$$

$$w^k = \begin{bmatrix} \nabla f_1(C\xi^k) \\ \nabla f_2(C\xi^k) \\ \vdots \\ \nabla f_n(C\xi^k) \end{bmatrix}$$

Finite-Sum Methods as Jump Systems

Choose $A_{i_k} = \tilde{A}_{i_k} \otimes I_p$, $B_{i_k} = \tilde{B}_{i_k} \otimes I_p$, and $C = \tilde{C} \otimes I_p$ where

Method	$\tilde{A}_{i_k}$	$\tilde{B}_{i_k}$	$\tilde{C}$
SAGA	$\begin{bmatrix} I_n - e_{i_k}e_{i_k}^T & \tilde{0} \\ -\frac{\alpha}{n}(e - ne_{i_k})^T & 1 \end{bmatrix}$	$\begin{bmatrix} e_{i_k}e_{i_k}^T \\ -\alpha e_{i_k}^T \end{bmatrix}$	$\begin{bmatrix} \tilde{0}^T & 1 \end{bmatrix}$
SAG	$\begin{bmatrix} I_n - e_{i_k}e_{i_k}^T & \tilde{0} \\ -\frac{\alpha}{n}(e - e_{i_k})^T & 1 \end{bmatrix}$	$\begin{bmatrix} e_{i_k}e_{i_k}^T \\ -\frac{\alpha}{n} e_{i_k}^T \end{bmatrix}$	$\begin{bmatrix} \tilde{0}^T & 1 \end{bmatrix}$
Finito	$\begin{bmatrix} I_n - e_{i_k}e_{i_k}^T & \tilde{0} \\ -\alpha(e_{i_k}e^T) & I_n - e_{i_k}(e_{i_k}^T - \frac{1}{n}e^T) \end{bmatrix}$	$\begin{bmatrix} e_{i_k}e_{i_k}^T \\ \tilde{0}\tilde{0}^T \end{bmatrix}$	$\begin{bmatrix} -\alpha e^T & \frac{1}{n}e^T \end{bmatrix}$
SDCA	$I_n - \alpha mne_{i_k}e_{i_k}^T$	$-\alpha mne_{i_k}e_{i_k}^T$	$\frac{1}{mn}e^T$

Sparsity of B_{i_k} captures the low cost of stochastic methods.

Quadratic Contraints

If g is L-smooth and m-strongly convex, then

$$\begin{bmatrix} x - x^* \\ \nabla g(x) \end{bmatrix}^T \begin{bmatrix} -2mLI_p & (m+L)I_p \\ (m+L)I_p & -2I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \nabla g(x) \end{bmatrix} \geq 0$$

Assumptions on f_i give

$$\begin{bmatrix} x - x^* \\ \nabla f_i(x) - \nabla f_i(x^*) \end{bmatrix}^T \begin{bmatrix} 2L\gamma I_p & (L-\gamma)I_p \\ (L-\gamma)I_p & -2I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \nabla f_i(x) - \nabla f_i(x^*) \end{bmatrix} \geq 0$$

LMI Conditions for Rate Analysis

- Numerically solving the analysis LMI

$$\frac{1}{n}\sum_{i=1}^{n} \begin{bmatrix} A_i^T P A_i - \rho^2 P & A_i^T P B_i \\ B_i^T P A_i & B_i^T P B_i \end{bmatrix} + \begin{bmatrix} C & D \end{bmatrix}^T M \begin{bmatrix} C & D \end{bmatrix} \prec 0$$

  reveals opportunities and difficulties with different methods.

- After implementing the LMI once, one then only needs to modify the (A_i, B_i, C) matrices for every new method.

Conclusions from Numerical Results

- For SAGA, the LMI is consistent with existing rate. It even suggests that we can use a diagonal Lyapunov function.
- For Finito, the LMI requires that we use Lyapunov functions with off-diagonal terms. Hence, we can tell in the early stage of our analysis that Finito is significantly more difficult than SAGA.
- For SAG, the LMI based on static quadratic bounds is not feasible. The inequalities used to formulate the LMI are too conservative! SAG requires the convexity of g! We need more advanced inequalities, for example the so-called weighted off-by-one IQC!

Outline

- Integral Quadratic Constraints
- Jump Dynamic Systems
- Numerical Rate Bounds for Stochastic Algorithms
- **Analytical Rate Bounds**
- Conclusions

Simplified Parameterization

Method	Parameterization of $\tilde{P}$	Matrix Form of the Resultant LMI
SAGA	$\begin{bmatrix} p_1 I_n & \tilde{0} \\ \tilde{0}^T & p_2 \end{bmatrix}$	$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & q_4 e & \mu_6 I_n + q_6 ee^T \\ q_4 e^T & \mu_2 & q_5 e^T \\ \mu_6 I_n + q_6 ee^T & q_5 e & \mu_3 I_n + q_3 ee^T \end{bmatrix}$
SDCA	$p_1 I_n + p_2 ee^T$	$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_3 I_n + q_3 ee^T \\ \mu_3 I_n + q_3 ee^T & \mu_2 I_n + q_2 ee^T \end{bmatrix}$
Finito	$\begin{bmatrix} p_1 I_n + p_2 ee^T & p_3 ee^T \\ p_3 ee^T & p_4 I_n + p_5 ee^T \end{bmatrix}$	$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_4 I_n + q_4 ee^T & \mu_6 I_n + q_6 ee^T \\ \mu_4 I_n + q_4 ee^T & \mu_2 I_n + q_2 ee^T & \mu_5 I_n + q_5 ee^T \\ \mu_6 I_n + q_6 ee^T & \mu_5 I_n + q_5 ee^T & \mu_3 I_n + q_3 ee^T \end{bmatrix}$

Simplified LMI for SAGA

Suppose i_k is uniformly sampled and $m > 0$. Let a testing rate $0 \leq \rho \leq 1$ be given. Suppose $g \in \mathcal{S}(m, L)$, and γ is defined based on assumptions on f_i. If there exist positive scalars p_1, p_2, and non-negative scalars λ_1, λ_2 such that

$$\begin{bmatrix} p_2\alpha^2 + \left(\frac{n-1}{n} - \rho^2\right) np_1 & -\alpha^2 p_2 \\ -\alpha^2 p_2 & p_1 + \alpha^2 p_2 - 2\lambda_2 \end{bmatrix} \leq 0$$

$$\begin{bmatrix} (1-\rho^2)p_2 - 2\lambda_1 mL + 2\lambda_2 L\gamma & -\alpha p_2 + (m+L)\lambda_1 + (L-\gamma)\lambda_2 \\ -\alpha p_2 + (m+L)\lambda_1 + (L-\gamma)\lambda_2 & p_1 + \alpha^2 p_2 - 2\lambda_2 - 2\lambda_1 \end{bmatrix}$$

Then SAGA initialized with any $x^0 \in \mathbb{R}^p$ and $y_i^0 \in \mathbb{R}^p$ satisfies

$$\mathbb{E}\left[\|x^k - x^*\|^2 + \frac{p_1}{p_2}\sum_{i=1}^{n}\|y_i^k - \nabla f_i(x^*)\|^2\right] \leq \rho^{2k} R^0$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{p_1}{p_2}\sum_{i=1}^{n}\|y_i^0 - \nabla f_i(x^*)\|^2$.

SAGA with Individual Convexity

When f_i is m-strongly convex, we have $\gamma = -m$, and the LMI becomes

$$\begin{bmatrix} p_2\alpha^2 + \left(\frac{n-1}{n} - \rho^2\right) np_1 & -\alpha^2 p_2 \\ -\alpha^2 p_2 & p_1 + \alpha^2 p_2 - 2\lambda_2 \end{bmatrix} \leq 0$$

$$\begin{bmatrix} (1-\rho^2)p_2 - 2(\lambda_1 + \lambda_2)mL & -\alpha p_2 + (m+L)(\lambda_1 + \lambda_2) \\ -\alpha p_2 + (m+L)(\lambda_1 + \lambda_2) & p_1 + \alpha^2 p_2 - 2\lambda_2 - 2\lambda_1 \end{bmatrix} \leq 0$$

We can choose $p_1 = \frac{1}{L}$, $p_2 = \frac{1}{\alpha}$, $\lambda_1 = 0$, and $\lambda_2 = \frac{1}{L}$ to show

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{2L\alpha - 1}{(L\alpha - 1)n}, 2m\alpha - \frac{\alpha m^2}{(1 - L\alpha)L}\right\}\right)^k R^0$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{\alpha}{L}\sum_{i=1}^{n}\|y_i^0 - \nabla f_i(x^*)\|^2$. Choosing $\alpha = \frac{1}{3L}$, we have standard SAGA rate $\rho^2 = 1 - \min\left\{\frac{1}{3n}, \frac{m}{3L}\right\}$.

SAGA without Individual Convexity

When f_i is only L-smooth (not necessarily convex), we have $\gamma = L$, and the LMI becomes

$$\begin{bmatrix} p_2\alpha^2 + \left(\frac{n-1}{n} - \rho^2\right) n p_1 & -\alpha^2 p_2 \\ -\alpha^2 p_2 & p_1 + \alpha^2 p_2 - 2\lambda_2 \end{bmatrix} \leq 0$$

$$\begin{bmatrix} (1-\rho^2)p_2 - 2\lambda_1 m L + 2\lambda_2 L^2 & -\alpha p_2 + (m+L)\lambda_1 \\ -\alpha p_2 + (m+L)\lambda_1 & p_1 + \alpha^2 p_2 - 2\lambda_2 - 2\lambda_1 \end{bmatrix} \leq 0$$

When $\alpha = \frac{m}{4(m^2 n + L^2)}$, we can choose $b = \frac{2(m^2 n + L^2)}{L^2}$, $p_1 = b\alpha > 0$, $p_2 = \frac{1}{\alpha}$, $\lambda_1 = \frac{1}{L} \geq 0$, and $\lambda_2 = b\alpha$ to show

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \frac{m^2}{8(m^2 n + L^2)}\right)^k R^0$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{m^2}{8(m^2 n + L^2)L^2} \sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. Hence, the ϵ-optimal iteration complexity of SAGA without individual convexity is $\tilde{O}\left(\left(\frac{L^2}{m^2} + n\right)\log\left(\frac{1}{\epsilon}\right)\right)$.

Conclusions from Simplified LMIs

- Finito and SDCA (with and without individual convexity) can be analyzed similarly.
- When assumptions on f_i change, we only need to modify the value of γ and solve the resultant LMI.
- The LMI for SAGA only has 4 decision variables!
- Finito requires off-diagonal terms in the Lyapunov function and the resultant LMI has 7 decision variables! We only prove $\tilde{O}\left(n \log\left(\frac{1}{\epsilon}\right)\right)$ under a big data condition $n \geq \frac{48L^2}{m^2}$.
- SAG requires advanced quadratic constraints, and the resultant LMI has 10 decision variables! Analytically hard! This explains why the original proof for SAG is involved!

Summary

- **Automate** rate analysis of stochastic finite-sum methods.
- Distinguish difficult methods, e.g. SAG, from easy methods, e.g. SAGA, at early stage.
- Use numerical semidefinite programs to support search for analytical proofs.

Bin Hu, Peter Seiler, and Anders Rantzer, “A unified analysis of stochastic optimization methods using jump system theory and quadratic constraints," Conference On Learning Theory, COLT 2017.

Future Work

- Analysis of **Acceleration**
- Automated Algorithm **Design**
- Worst case analysis from dual problem
- Asynchronous Settings and Time Delays